Theses and Dissertations | 1. Thesis and Dissertation Collection, all items

1999-03

Time difference of arrival (TDOA) estimation using wavelet based denoising

Aktas, Unal.

Monterey, California: Naval Postgraduate School

http://hdl.handle.net/10945/13574

NAVAL POSTGRADUATE SCHOOL
Monterey, California

UNITED STATES NAVY
★ NAVAL POSTGRADUATE SCHOOL ★

19990310 015

THESIS

TIME DIFFERENCE OF ARRIVAL
(TDOA) ESTIMATION USING
WAVELET BASED DENOISING

by

Unal Aktas

March 1999

Thesis Advisor:	Ralph D. Hippenstiel
Co-Advisor:	Tri T. Ha

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE March 1999	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE TIME DIFFERENCE OF ARRIVAL (TDOA) ESTIMATION USING WAVELET BASED DENOISING		5. FUNDING NUMBERS	
6. AUTHOR(S) Unal Aktas			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000		8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.		12b. DISTRIBUTION CODE	

13. ABSTRACT *(maximum 200 words)*

The localization of mobile wireless communication units is studied. The most important method of localization is the time difference of arrival (TDOA) method. The wavelet transform is used to increase the accuracy of TDOA estimation. Several denoising techniques based on the wavelet transform are presented in this thesis. These techniques are applied to different types of test signals as well as the GSM signal. The results of the denoising techniques are compared to the ones obtained using no denoising. The denoising techniques allow a 28 to 81 percent improvement in the TDOA estimation.

14. SUBJECT TERMS. Time Difference of Arrival, Wavelet, Denoising, GSM			15. NUMBER OF PAGES 119
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18 298-102

TIME DIFFERENCE OF ARRIVAL (TDOA) ESTIMATION USING WAVELET BASED DENOISING

Unal Aktas
Lieutenant Junior Grade, Turkish Navy
B.S., Turkish Naval Academy, 1993

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
March 1999

Author:
_____
Unal Aktas

Approved by:
_____
Ralph D. Hippenstiel, Thesis Advisor

_____
Tri T. Ha, Co-Advisor

_____
Jeffrey B. Knorr, Chairman
Department of Electrical and Computer Engineering

iii

ABSTRACT

The localization of mobile wireless communication units is studied. The most important method of localization is the time difference of arrival (TDOA) method. The wavelet transform is used to increase the accuracy of TDOA estimation. Several denoising techniques based on the wavelet transform are presented in this thesis. These techniques are applied to different types of test signals as well as the GSM signal. The results of the denoising techniques are compared to the ones obtained using no denoising. The denoising techniques allow a 28 to 81 percent improvement in the TDOA estimation.

TABLE OF CONTENTS

xi

I. INTRODUCTION

The problem of providing reliable and accurate position information of mobile wireless communication units, has attracted a lot of attention in recent years. The main factor behind the recent interest has been the adoption of certain regulations by the Federal Communications Commission (FCC) [1], that require wireless communications licensees to incorporate a position localization capability in their systems to provide Enhanced-911 (E-911) service. However, there are many other reasons for wireless service providers to have such a system in place. For example, one can use reliable position localization as means to optimize the performance and design of the wireless networks.

A.   THESIS OUTLINE

In the remainder of this Chapter, reasons for localizing cellular emitters and standard methods to accomplish this task are explored. Chapter II introduces the GSM system. Chapters III-VI discuss wavelet based denoising methods. Test signal description and simulation results are contained in Chapter VII. Chapter VIII contains the conclusion and suggests logical extension of the work.

B.   THESIS CONTRIBUTIONS

The main contribution of this work is the reduction in mean square error for the time difference of arrival estimate. This permits localization with corresponding smaller error. The novel denoising methods, a modified approximate maximum likelihood (MAML), a fourth order moment, and a time varying MAML method, provide desirable improvements.

1

C.   WHY IS THERE A NEED TO LOCATE CELLULAR PHONES?

1.   Public Safety and Enhanced Emergency Services

The FCC of the United States passed a regulation to provide E-911 service whereby "carriers are required to identify the latitude and longitude of a mobile unit making a 911 call." At present, if a cellular subscriber dials 911 and does not relay location information to the operator, the authorities can only estimate the caller's location to within a few kilometers, based on the cell being used. Providing location and tracking to emergency services would save critical seconds in response time to stranded motorists, injury victims/witnesses who are confused or unable to relay geographical information, or allow vehicle pursuit by authorities.

2.   Fleet/Asset Management for Couriers and Transportation Businesses

Many organizations already utilize cellular phones for their day-to-day business activities. It makes sense to utilize these phones, or wireless transmitters using cellular technology, to monitor and track vehicles and shipments, including couriers, taxis, buses, and other fleet-based commercial services.

3.   Tracking of Stolen Phones and/or Criminals

Stolen and cloned cellular and personal communications systems (PCS) phones represent a major problem for cellular carriers and users. Authorities acknowledge an increased use of mobile phones in criminal activities. Until now, mobile phones used in illegal activities have been difficult to trace and millions of dollars in wireless fraud have been committed. With the availability of cellular phone tracking and localization, the criminal element in wireless phone use could diminish dramatically.

4. Tracking Stolen Vehicles

Many vehicles are equipped with a cellular phone, and many vehicles are being manufactured with a tracking "tag" for location or navigational purposes. The localization technology and its future variations offer the ability of recovering lost or stolen vehicles.

5. Location-Sensitive Navigation

Imagine driving into new and unfamiliar city, and being able to dial up navigational service. The operator could identify your current location and give you directions to the nearest service station or hotel based on your present position.

6. Localization of Cellular and PCS Telephone Users

Similarly to the navigational service, a pay-per-use service may allow tracking of persons who use cell phones. Parents who need to locate their children or family members wanting to find a medical patient will benefit from this technology, saving time and reduce worry. If one is uncertain about his own location, this service could help him to give position or address information.

7. Electronic Databases

Often, it is useful to study the demographics of an area to determine the necessity for roads and other infrastructures. By tracking the use and location of cellular phone users, cellular carriers can strategically plan base stations.

8. Law Enforcement or Military Use

Position information can also be made available to law enforcement and military users. The information allows localization of a user independent of his desire to do so.

D. EXISTING POSITION LOCALIZATION SYSTEMS

A number of position localization systems have evolved over the years. These position localization systems are: *Global Positioning System (GPS), Loran C, SIGNPOST Navigation, Global Navigation Satellite System, Automatic Vehicle Monitoring,* and *Cellular Phones Geolocation.*

1. Global Positioning System (GPS)

GPS is the most popular radio navigation aide, due to high accuracy, worldwide availability, and low cost. GPS is also used to relay position of cellular phones to *public switched telephone networks* (PSTN) or to *public safety answering points* (PSAP).

The principle behind GPS is very simple. GPS uses a time-of-arrival (TOA) method. GPS uses precise timing within a group of satellites that transmit a spread spectrum L-band signal to ground in the centered at 1575.42 MHz. An accurate clock at the receiver measures the time delay between the signals leaving the satellite and arriving the receiver. This allows calculation of the distance from the satellite to the cellular phone. If one obtains three distances using three different satellites, one can use a triangulation method to determine the position of the cellular phone.

Currently a GPS receiver costs less than $200, and its accurate to 100 meters [13, 23]. More sophisticated units, including those used by military, which use differential GPS, provide an accuracy of a few meters.

2. Loran C

Loran C is a navigational tool that operates in the low frequency (90-110 kHz) band and uses a pulsed hyperbolic system for triangulation. It has a repeatable accuracy in the 19-90 meters range and is accurate to about 100 meters with 95 percent confidence

and 97 percent availability. Like GPS, its performance depends on local calibration and topography. GPS has replaced Loran C in most applications [13].

3. SIGNPOST Navigation

SIGNPOST Navigation employs a large number of simple radio transmitters to accurately determine position of a mobile. These transmitters are replaced along highways and typically serve as coded beacons, where the code designates the latitude and longitude of the SIGNPOST. The transmitter strength indicates the relative position of the receiver to the transmitter. This navigation aids work well for limited areas such as a small city. While not originally designated as such, today's Advanced Mobile Phone System (AMPS) analog cellular radio system may actually serve as a SIGNPOST system, since each base system transmits a beacon signal on its forward control channel [24]. As a part of the forward control channel structure, an overhead message containing a station identification number (SID) and a digital color code (DCC) is sent every 0.8 seconds. The DCC is used to determine the location within the cell [13].

4. Global Navigation Satellite System (GLONASS)

The Global Navigation Satellite System (GLONASS) is similar system to GPS. Although the system uses principles similar to GPS, its operation differs in several aspects. The synchronization period of GLONASS takes only 1/3 as long as GPS, typically under a minute [13].

5. Automatic Vehicle Monitoring (AVM)

Automatic Vehicle Monitoring (AVM) systems provide position localization capabilities for handling a large number of vehicles. Typical applications include fleet management, vehicle security, and emergency services. AVM systems have been

available in the United States for a number of years [13]. In 1995, the FCC changed the name of these systems to *Location and Monitoring Services (LMS)*. In the United States, the primary band for LMS is the 902-928 MHz industrial, scientific, and medical (ISM) band. LSM is also supported to a lesser extent in several bands below 512 MHz. The LSM system is a licensed system with up to 300 W peak power for the forward link; however it shares the band with low-power unlicensed devices, such as cordless phones, wireless local area networks, and utility meter reading systems. The band is also used by federal government radiolocation systems and amateur radio operators, so the prospect of the interference between LMS and other users of the spectrum is an issue in the deployment of LMS systems [25].

E.   METHODS FOR LOCATING CELLULAR PHONES

There are many techniques that can be considered for the localization of cellular phones. These techniques can be classified into two categories. These are, position localization systems that require a modification of the existing handsets, and systems that require modification at the base stations. The modification of existing handsets can be accomplished by using the GPS based position localization system, mobile assisted time or time difference of arrival techniques. The second category consists of *angle of arrival* (AOA), *time difference of arrival* (TDOA), *time of arrival* (TOA), or *frequency difference of arrival* (FDOA) estimation of the wavefront at the receiving platforms.

1.   Angle of Arrival (AOA)

AOA can be accomplished by means of a highly directive receiving antenna or by means of a nulling measurement using several feeds from the antenna. A single platform may be sufficient for AOA position localization of a wireless transmitter on the surface of

the earth. Once the angle is precisely determined, the position of the emitter can be determined by the intersection of the centerline of the antenna beam, the *boresight,* with the surface of the earth.

Although AOA methods offer a practical solution for the wireless transmitter localization, they have certain drawbacks. For accurate AOA estimates, it is crucial that signals coming from the source to antenna arrays must be coming from the *Line-Of-Sight* (LOS) direction. However, this is often not the case in cellular systems, which may be operating in heavily shadowed channels, such as those encountered in urban environments. Another factor is the considerable cost of installing antenna arrays. This method also requires a relatively complex AOA algorithm. Although there are exceptions, these algorithms tend to be highly complex because of the need for measurement, storage and usage of array calibration data and their computationally intensive nature [3, 13].

2. Frequency Difference of Arrival (FDOA)

FDOA measurements require at least two receiving platforms and that the relative velocity between the platforms be large enough that the difference in Doppler shifts of the two received signals is significantly larger than the frequency measurement error.

3. Time of Arrival (TOA)

It may be possible for the base station to indirectly determine the time that the signal takes from the source to the receiver on the forward or the reverse link. This may be done by measuring the time in which the mobile responds to an inquiry or an instruction transmitted to the mobile from the base station. The total time elapsed from the instant the command is transmitted to the instant the mobile response is detected, is

7

composed of the sum of the round trip signal delay and any processing and response delay within the mobile unit. If the processing delay of the response within the mobile is known with sufficient accuracy, it can be subtracted from the total measured time, which provides us the total round trip delay.

There are certain problems with this method. The estimate of the response delay within the mobile might be difficult to determine in practice. The main reason is the variations in designs of the handsets from different manufacturers. Secondly, this method is highly susceptible to timing errors in the absence of LOS, as there would be no simple way to reduce the errors induced by multiple signal reflections on the forward or reverse link.

4. Time Difference of Arrival (TDOA)

The classical approach to estimating TDOA is to compute the cross correlation between signals arriving at two base stations. The TDOA estimate is taken as the delay, which maximizes the cross-correlation function. The cross-correlation function is also used to determine at which base station the signal arrives first. These two pieces of information yield a hyperbolic localization curve. We can localize the wireless transmitter by solving two hyperbolic curve equations.

It is necessary that the code generators at each receiver be synchronized so that the TDOA estimates have a common time base [2]. This form of radio localization is useful in asynchronous system since time of transmission need not be known. In geometric interpretation, this procedure reduces to finding the intersection of hyperbolas whose foci are at the receivers. To determine the location of a transmitter in two dimensions, at least three receivers are required.

8

This method offers many advantages over other competing techniques. No modification of the existing handsets is required. In this respect, this solution would be more cost effective than a GPS-based solution. It also does not require knowledge of the absolute time of transmission from the handset like a modified TOA method needs. Since this technique does not require any special type of antennas, it is less expensive to put in place than the AOA methods. It can also provide some immunity against timing errors if the source of major signal reflections is near the mobile. If a major reflector effects the signal components going to all the receivers, the timing error may get cancelled or reduced in the time difference operation. Hence, TDOA methods may work accurately in some situations where there is no LOS signal component. In this respect, it is superior to the AOA method and TOA methods.

II. GLOBAL SYSTEM FOR MOBILE (GSM)

GSM is a second-generation cellular system standard that was developed to solve the fragmentation problems of Europe's first cellular systems. GSM is the first cellular system to specify digital modulation, network level architectures and services. Before GSM, European countries used different cellular standards throughout the continent, and it was not possible to use a given single subscriber unit throughout Europe. GSM was originally developed to serve as the Pan-European cellular service and promised a wide range of network service through the use of the *Integrated Services Digital Network (ISDN)*. GSM's success has exceeded the expectations of virtually everyone, and it is now the world's most popular standard for new cellular radio and personal communications equipment.

A.   GSM SYSTEM ARCHITECTURE

The GSM system architecture consists of three major interconnected subsystems that interact with themselves and the users through network interfaces. The subsystems are the *Base Station Subsystem* (BSS), *Network and Switching Subsystem* (NSS), and the *Operation Support Subsystem* (OSS). *The Mobile Station* (MS) is also a subsystem, but is usually considered to be part of the BSS for architectural purposes.

The BSS provides and manages radio transmission between the MS's and the Mobile Switching Center (MSC). The BSS also manages the radio interface between the MS's and all other subsystems of GSM.

11

The NSS manages the switching functions of the system and allows the MSC's to communicate with other networks such as the *Public Switched Telephone Network* (PSTN) and *Integrated Services Digital Network* (ISDN).

The OSS supports the operation and the maintenance of GSM and allows system engineers to monitor, diagnose, and troubleshoot all aspects of GSM. This subsystem interacts with the other GSM subsystems.

1. GSM Signal Specifications

GSM utilizes two 25 MHz bands, which have been set aside for system use in all member countries. The 890-915 MHz band is used for subscriber-to-base transmission (reverse link), and the 935-960 MHz band is used for base-to-subscriber transmission (forward link). GSM uses *Frequency Division Duplex* (FDD) and a combination of *Time Division Multiple Access* (TDMA) and *Frequency Hopped Multiple Access* (FHMA) schemes to provide stations with simultaneous access to multiple users. The available forward and reverse frequency bands are divided into 200 KHz channels. These channels are identified by their *Absolute Radio Frequency Channel Number* (ARFCN). The ARFCN denotes a forward and reverse channel pair, which is separated in frequency by 45 MHz. Each channel is time shared between as many as eight subscribers using TDMA.

The radio transmissions on both forward and reverse link are made at a channel data rate of 270.833 Kbps using binary *0.3 GMSK* modulation. The bandwidth-bit duration product, *BT*, has a level of 0.3. The signaling bit duration is 3.692 μs. User data is sent a maximum rate of 24.7 Kbps. Each time slot (TS) has an equivalent time allocation allowing for 156.25 channel bits. From this, 8.25 bits of guard time and 6 total

start and stop bits are used to prevent overlap with adjacent time slots. Each TS has a time duration of 576.92 μs, while a single GSM TDMA frame spans 4.615 ms. The total number of available channels within a 25 MHz bandwidth is 125. Table 2.1 summarizes the signal specifications.

Parameter	Specifications
Reverse Channel Frequency	890-915 MHz
Forward Channel Frequency	935-960 MHz
ARFCN Number	0 to 124 and 975 to 1023
Tx/Rx Frequency Spacing	45 MHz
Tx/Rx Time Slot Spacing	3 Time slots
Modulation Data Rate	270.833333 Kbps
Frame Period	4.615 ms
Users per Frame (Full Rate)	8
Time Slot Period	576.9 μs
Bit Period	3.692 μs
Modulation	0.3 GMSK
ARFCN Channel Spacing	200 KHz
Interleaving (max delay)	40 ms
Voice Coder Bit Rate	13.4 Kbps.

Table 2.1: GSM signal specifications.

2.   Gaussian Minimum Shift Keying (GMSK)

GMSK is a binary modulation scheme which may be viewed as a derivative of *Minimum Shift Keying* (MSK). In GMSK, the sidelobe levels of the spectrum are reduced by passing the modulating *Non-return to zero* (NRZ) data waveforms through a pre-modulation Gaussian pulse-shaping filter. Baseband Gaussian pulse shaping smoothes the phase trajectory of the MSK signal and hence stabilizes the instantaneous frequency variations over time. This has the effect of considerably reducing the sidelobe levels in the transmitted spectrum.

3. GSM Channel Types

There are two types of GSM logical channels, called *traffic channels* (TCH) and *control channels* (CCH). Traffic channels carry digitally encoded user speech or user data and have identical functions and formats on both the forward and reverse link. Control channels carry signaling and synchronizing commands between the base station and the mobile station.

4. Frame Structure for GSM

As shown in Figure 2.1, there are eight time slots (TS) per GSM frame. The frame period is 4.615 ms. A TS consists of 148 bits which are transmitted at rate of 270.833 Kbps (an unused guard time of 8.25 bit period is provided at the end of each burst). Out of the total 148 bits per TS, 114 are information bits, which are transmitted as two 57 bit sequences close to the beginning and end of the burst. A 26 bit training sequence allows the adaptive equalizer in the mobile or base station receiver to analyze the radio channel
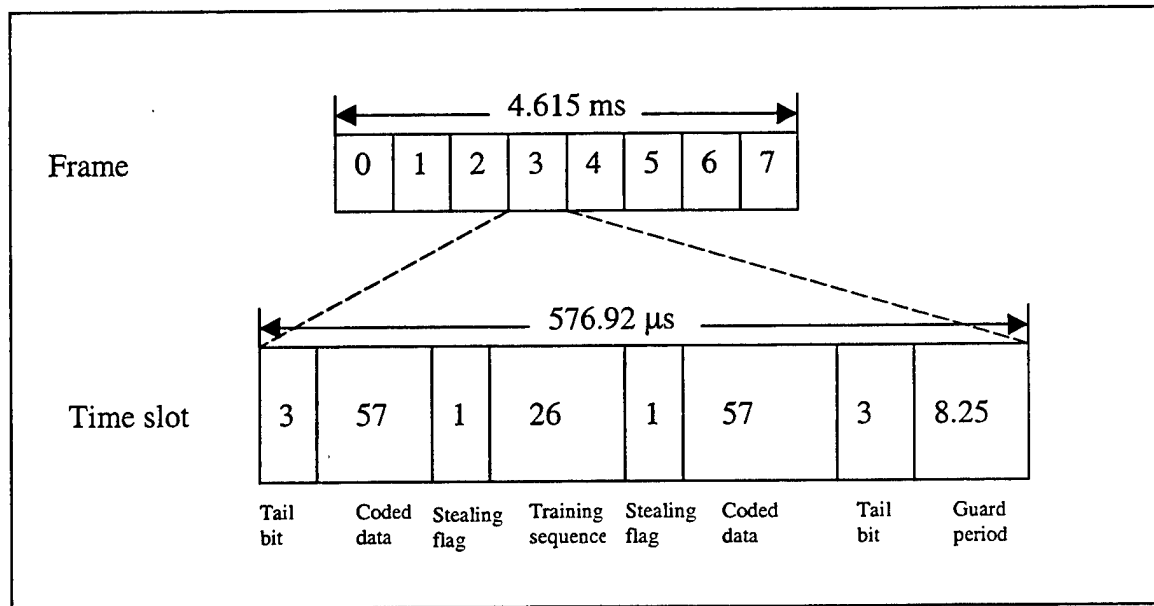
Figure 2.1: GSM frame structure.

14

characteristics before decoding the user data. Stealing flags on the both side of the training sequence are used to distinguish whether the TS contains voice or control data.

B.   TRANSMITTER/RECEIVER SYSTEM

In this section we will look at the GSM transmitter and receiver and simulate some GSM signals to be used for simulation purposes. The overall structure of the GSM transmitter/receiver system is illustrated in Figure 2.2.
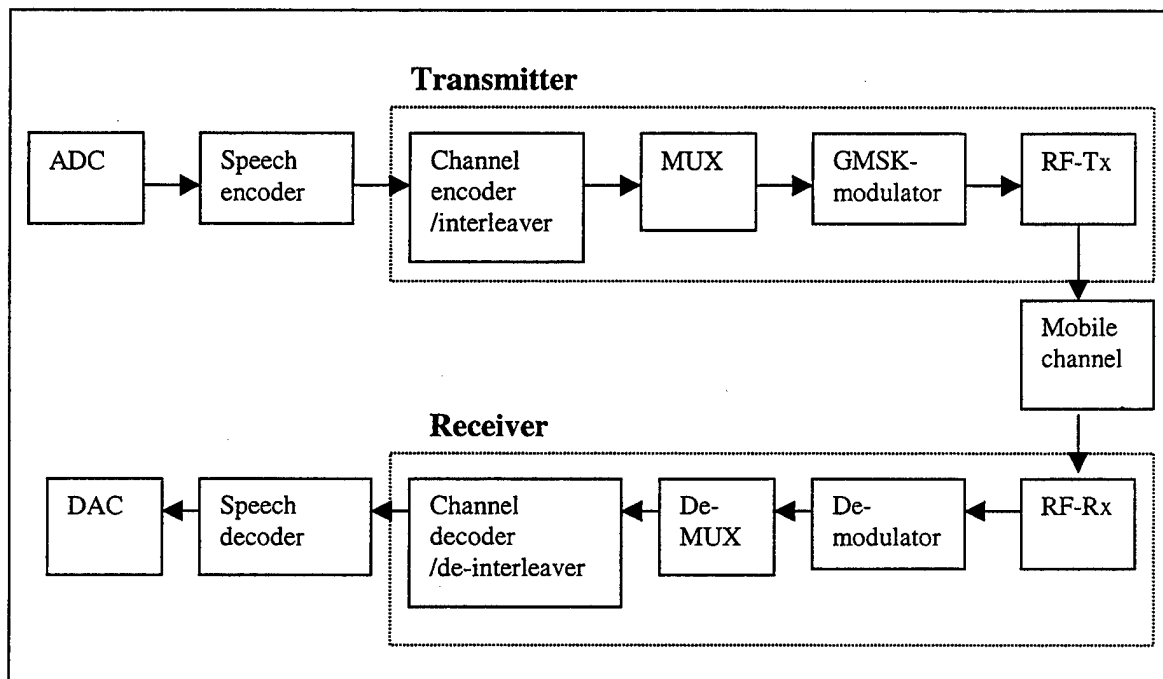
Figure 2.2:Block diagram for the GSM transmitter/receiver system.

1.   Transmitter Structure

The overall structure of the transmitter is illustrated in Figure 2.3. The transmitter is made up for distinct functional blocks.
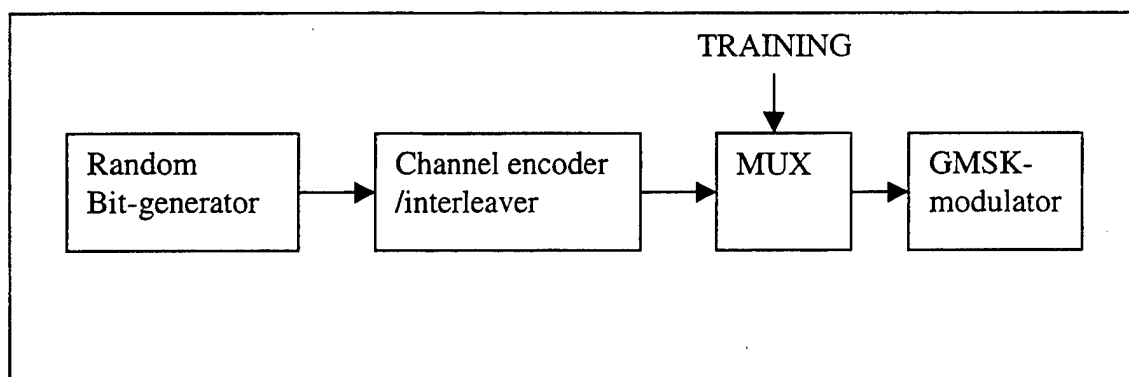
Figure 2.3: The overall structure of the transmitter.

To simulate an input data stream to the channel encoder/interleaver, a sequence of random data bits is generated by the random bit generator. This sequence is accepted by the MUX, which splits the incoming sequence to form a GSM normal burst. The burst type requires that a training sequence is included which also must be supplied. Upon having generated the prescribed GSM normal burst data structure, the MUX sends this to the GMSK-modulator, where GMSK is short for *Gaussian Minimum Shift Keying*. The GMSK-modulator block performs a differential encoding of the incoming burst to form a NRZ sequence. This modified sequence is then subject to the actual GMSK-modulation after which, the resulting signal is represented as a complex baseband I and Q signal.

2. Receiver Structure

The general structure of the receiver, consisting of three functional blocks, is illustrated in Figure 2.4. The demodulator accepts the GSM burst, r, using a complex baseband representation. Based on this data sequence, the oversampling rate *OSR,* the training sequence *TRAINING* , and the desired length of the receiving filter, *Lh*, the demodulator determines a bit sequence. This demodulated sequence is then used as the

input to the demultiplexer (DeMUX) where the actual data bits are obtained. The remaining control bits and the training sequence are stripped off. Finally channel decoding and de-interleaving is performed.
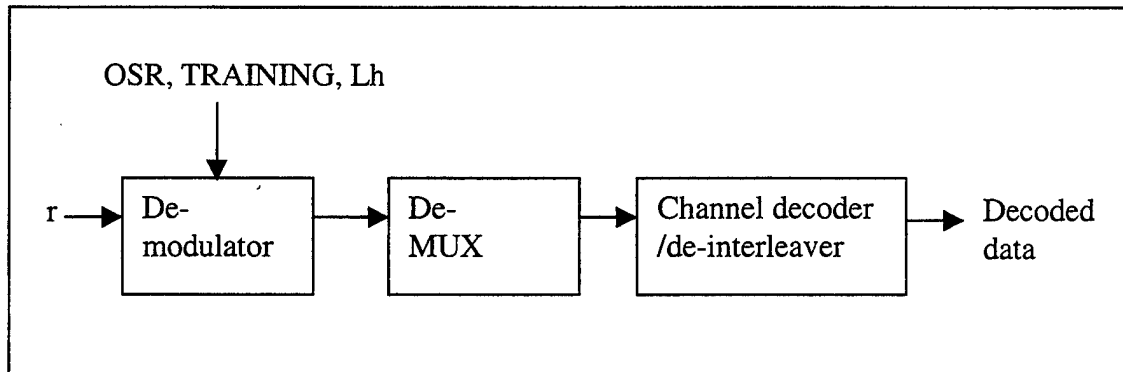
Figure 2.4: The overall structure of the receiver.

III. WAVELETS

Usually signals are transformed to obtain information that is not directly observable in the raw signal. There are many transformations that could be used. The wavelet transform belongs to this set. It provides a time-scale representation [6]. There are other transformation, which can give similar information, such as the short time Fourier transform, Wigner-Ville distribution, etc.

Wavelet analysis can be interpreted as an extension of the Fourier analysis. Thus, this chapter will first discuss Fourier and then present the wavelet analysis.

A.   FOURIER ANALYSIS

Fourier analysis breaks a signal into its constituent sinusoidal components at different frequencies. One can also think of the Fourier analysis as a mathematical technique for transforming the signal from a time-based to a frequency-based representation [7].

1.   Fourier Series

Let $g_p(t)$ denote a periodic signal with period T_0. By using the Fourier series expansion of this signal, we can resolve this signal into an infinite sum of complex exponentials. The Fourier series expansion can be written as [8]:

$$g_p(t) = \sum_{n=-\infty}^{\infty} c_n e^{j2\pi n f_0 t} \qquad , \qquad (3.1)$$

where n/T_0 represents the nth harmonic of the *fundamental frequency* $f_0 = 1/T_0$. The series expansion of Equation 3.1 is referred to as the *complex exponential Fourier series*. The c_n coefficients are the *complex Fourier coefficients*. We can determine c_n as follows:

$$c_n = \frac{1}{T_0} \int_{-T_0/2}^{T_0/2} g_p(t) e^{-j2\pi n f_0 t} dt \, , \quad n = 0, \pm 1, \pm 2, \cdots \, . \qquad (3.2)$$

The complex exponential analysis equation provides the coefficients necessary to reconstruct the periodic signal from its Fourier series coefficients. A plot of the magnitude of c_n versus frequency is called the magnitude spectrum of the signal $g_p(t)$. The spectrum provides frequency information of the signal.

2.   Fourier Transform

In the previous section we used the Fourier series to represent a periodic signal. We will develop a similar representation for a signal $g(t)$ that is nonperiodic. In order to do this, we first construct a periodic function $g_p(t)$ of period of T_0 in such a way that $g(t)$ defines one cycle of this periodic function. In limit we let the period T_0 become infinitely large, so that we may write

$$g(t) = \lim_{T_0 \to \infty} g_p(t) \, . \qquad (3.3)$$

The Fourier transform of a general continuous function $g(t)$ is defined as [8]:

$$G(f) = \int_{-\infty}^{\infty} g(t) e^{-j2\pi f t} dt \, . \qquad (3.4)$$

$G(f)$ is a continuous function of the frequency variable f. The original signal $g(t)$ can be recovered exactly from $G(f)$ by means of the inverse Fourier transform which is defined as follows [8]:

$$g(t) = \int_{-\infty}^{\infty} G(f) e^{j2\pi f t} df \, . \qquad (3.5)$$

The two functions $g(t)$ and $G(f)$ uniquely define each other and are known as a *Fourier transform pair*.

Fourier analysis is extremely useful because the signal's frequency content is of great importance. So why do we need another techniques, such as the wavelet analysis?

Fourier analysis has a serious drawback. In transforming to the frequency domain, time information is lost. When looking at a Fourier transform of a signal, it is impossible to tell when a particular event took place. If a signal is *stationary*, this drawback is not very important. However many interesting signals contain numerous non-stationary or transitory characteristics such as trends, abrupt changes, and beginnings and ends of events. These characteristics can be the most important part of the signal, and Fourier analysis is not suited to localize them.

3. Short-Time Fourier Analysis

The Fourier analysis technique described above provides a frequency domain presentation of the signal. When the signal is non-stationary, it is desirable to have a description that involves both time and frequency.

The short-time Fourier transforms (STFT) can be viewed as an extension of the Fourier transform devised to map the signal into the two dimensional time-frequency plane. The STFT uses a sliding window function $w(t)$ to segment the signal into small uniform blocks of time. Each block is made short enough so that the signal may be considered stationary within the segment. The Fourier transform is then applied to each segment given by;

$$S(\tau, f) = \int_{-\infty}^{\infty} g(t) w^*(t - \tau) e^{-j2\pi ft} dt, \tag{3.6}$$

where $w^*(t - \tau)$ denotes the sliding window, * represents the conjugation and $S(\tau, f)$ displays the evolution of the signal's frequency over time. The STFT represents a compromise between the time-and frequency-based views of a signal. It provides

21

information about time and spectral behavior of a signal. However one can only obtain this information with a fixed precision, where the precision is determined by the window. Many different window functions, *w(t)*, may be used. The choice will effect the characteristics of STFT. Once a window function has been chosen its shape and length will determine the resolution in time and in frequency. As a result of the uncertainty principle, the time resolution (Δt), and the frequency resolution (Δf) of a given signal are inversely related. Their product has lower bound of $1/4\pi$, which is achieved by the Gaussian window [9]. This produces a trade off of time resolution for frequency resolution and vice versa. Since the choice of window will fix (Δt) and (Δf) for the entire time axis, the STFT partitions the time-frequency plane into a uniform grid. The window can not simultaneously provide good time resolution (requiring short windows) and good frequency resolution (requiring long windows). The performance (i.e., frequency resolution) of a window is governed by its functional form [22].

B.   WAVELET ANALYSIS

1.   Introduction

A wavelet is defined as an oscillatary function of time or space [10]. The term wavelet comes from the French word ondelette, which means small wave. It has its energy concentrated in time, which allows the analysis of transient, non-stationary, or time-varying phenomena. We will take a wavelet transform and use it in the series expansion of signals similar to the Fourier series approach.

2.   The Continuous Time Wavelet Transform (CTWT)

The Fourier transform of *g(t)* is defined as:

$$G(f) = \int_{-\infty}^{\infty} g(t)e^{-j2\pi ft}dt \ .$$

(3.7)

This is the integral over all time of the signal *g(t)* multiplied by a complex exponential. Similarly, the continuous time wavelet transform (CTWT) is defined as the integral over all time of the signal multiplied by scaled, shifted versions of a wavelet function $\Psi(t)$:

$$C(\tau,a) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} g(t)\,\Psi^*\left(\frac{t-\tau}{a}\right) dt , \qquad (3.8)$$

where $\Psi(t)$ is the wavelet function. The parameter τ denotes translation in time, and the scale factor a denotes dilation or compression in time. The factor $1/\sqrt{a}$ normalizes the energy of the CTWT and * denotes conjugation.

One advantage of the wavelet analysis is that it allows the selection of large number of basis functions, contrary to being restricted to sinusoids as in the Fourier analysis. Two important characteristics of the wavelets are that; the wavelet function $\Psi(t)$ is of finite duration, the wavelet function $\Psi(t)$ has a zero mean and is zero at the end points. The first and second characteristic requires that the basis function oscillates about zero, and gives rise to the name wavelet or small wave [10].

The time resolution and frequency resolution of the CTWT is controlled by the scale factor a (Equation 3.8). Low scales (small values of a) correspond to high frequency wavelets and provide good time resolution. High scales (large values of a) correspond to low frequency wavelets with poor time resolution but good frequency resolution.

A second advantage of the wavelet analysis is the multi-resolution capability it provides in time-scale plane. The time-frequency mapping of the STFT and the CWT is shown in Figure 3.1
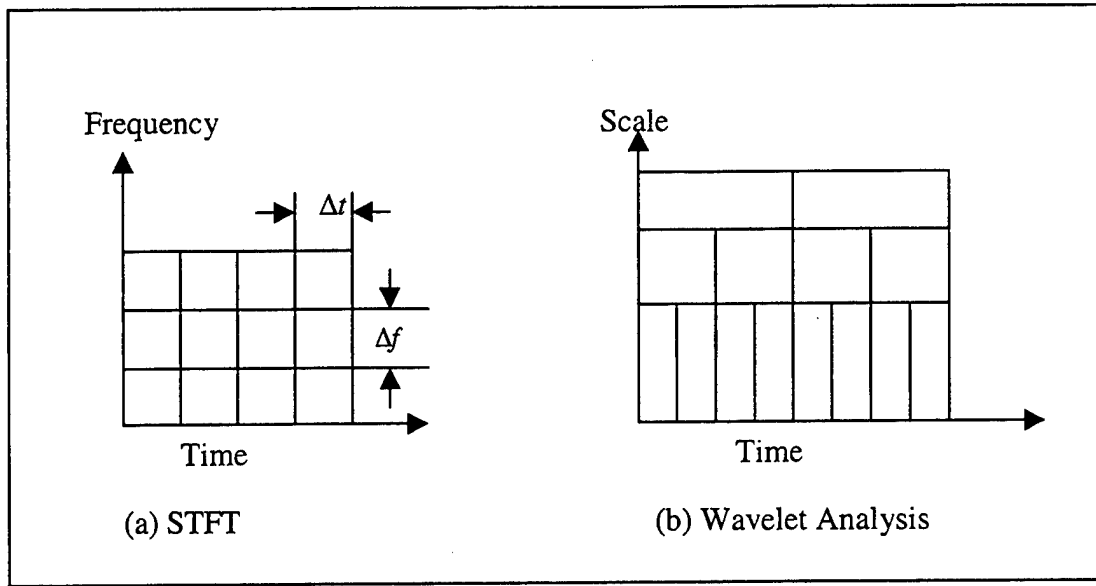
Figure 3.1: (a) Frequency-Time plane for STFT, (b) Scale-Time plane for CWT.

The STFT produces a uniform grid with a constant time (Δt) and frequency resolution (Δf), while the CTWT has a time and frequency resolution that depends on the scale. Note that the CTWT has a time resolution that improves at higher frequency while frequency resolution degrades.

3. The Discrete Wavelet Transform (DWT)

Although the discretized continuous wavelet transform enables the computation of the continuous wavelet transform by computers, it is not a true discrete transform [6]. As a matter of fact, the wavelet coefficients are simply a sampled version of the CWT, and the information it provides is highly redundant, as far as the reconstruction of the signal is concerned. This redundancy, on the other hand, requires a significant amount of computation time and resources.

The DWT is sufficient for most practical applications for the reconstruction of the signal. The DWT provides sufficient information, and can offer a significant reduction in the computation time. It is considerably easier to implement than the continuous wavelet transform and obtained by restricting the scale and time parameters of the CWT to discrete values. The DWT of a discrete signal g(n) is defined by

$$C(a,b) = \sum_n \frac{1}{\sqrt{a}} g(n)\Psi^*\left(\frac{n-b}{a}\right), \qquad (3.9)$$

where a, b, and n are the discrete versions of a, τ, and t of Equation 3.8 respectively. The scaling factor is further restricted to;

$$a = a_0^J, \qquad J=0,1,\ldots \log_2(N). \qquad (3.10)$$

The choice of a_0 will govern the accuracy of the signal reconstruction via the inverse transform. It is popular to choose $a_0=2$, since it permits the implementation of fast algorithms. Setting $a = 2^J$ produces octave bands called dyadic scales. As the scale level is increased from J to J+1, the analysis wavelet is stretched in the time domain by a factor of two. Hence the DWT output has better frequency resolution and less precise time resolution as the scale number increases.

If the time shifting parameter b is restricted to $k2^J$, where k is an integer, this version of the DWT is known as the decimated DWT and can be written as:

$$C(2^J, k2^J) = \sum_{n=1}^{N} \frac{1}{\sqrt{a}} g(n)\Psi^*\left(2^{-J}n-k\right); \qquad (3.11)$$

where $J = 0, 1, 2, \cdots, \log_2(N)$, $k = 1, 2, \cdots, N2^{-J}$, and N is the length of the signal $g(n)$. The term $k2^J$ in the argument of DWT, indicates that $C(a, b)$ is decimated by a factor of two at each successive scale J by retaining only the even points.

25

a. Subband Coding and Multiresolution Analysis

The time-scale (frequency) representation of the signal is obtained by using digital filtering techniques. The CWT is computed by changing the scale of the analysis window, shifting the window in time, multiplying by the signal, and integrating over all times. In the discrete case, filters of different cutoff frequencies are used to analyze the signal at different scales. The signal is passed through a series of high pass filters, and passed through a series of low pass filters. Each one of the filter is followed by a two-to-one decimator.

The DWT analyzes the signal at different frequency bands with different resolutions by decomposing the signal into a coarse and a detail component at each scale. The DWT employs two sets of functions, a scaling function and a wavelet function. These can be associated with a lowpass and a highpass filter, respectively. The decomposition of the signal into different frequency bands is obtained by successive highpass and lowpass filtering of the signal followed by the decimation operation.

Figure 3.2 illustrates this procedure. Here g[n] is the original signal to be decomposed, and d[n] and h[n] are the lowpass and highpass filters respectively. The bandwidth of the signal at every level is marked on the figure as BW. The original signal is first passed through a halfband highpass filter h[n] and a lowpass filter d[n]. After filtering, half of the samples can be eliminated according to the Nyquist's rule. This is because the output has a high frequency of $\pi/2$ radians instead of π. The signal can be subsampled, by simply discarding every other sample. This decomposition halves the time resolution, since the output is characterized by half the number of samples compared to the original signal. However this operation doubles the frequency resolution, since the

frequency band of the signal now spans only half the input frequency band. The above procedure, which is known as the subband coding, is repeated.

C.   THE EFFECTINESS OF WAVELET ANALYSIS

Wavelet expansions and wavelet transforms have been shown to be very efficient and effective in analyzing a very wide class of signals and phenomena [10].

1.   The wavelet expansion allows a more accurate time description and identification of signal characteristics. A Fourier coefficient represents a component that lasts for the integration time of the transform and, therefore, temporary events must be described by a phase characteristic that allows cancellation or reinforcement over large time periods. A wavelet expansion coefficient represents a component that is itself local and is easier to interpret. The wavelet expansion may allow a separation of components of a signal that overlap in time or frequency.

2.   Wavelets are adjustable and adaptable. Because there is not just one wavelet, they can be designed to fit individual applications. They are ideal for adaptive systems that adjust themselves to accommodate the signal.

3.   The generation of wavelets and the calculation of the discrete wavelet transform are well suited for digital implementation.
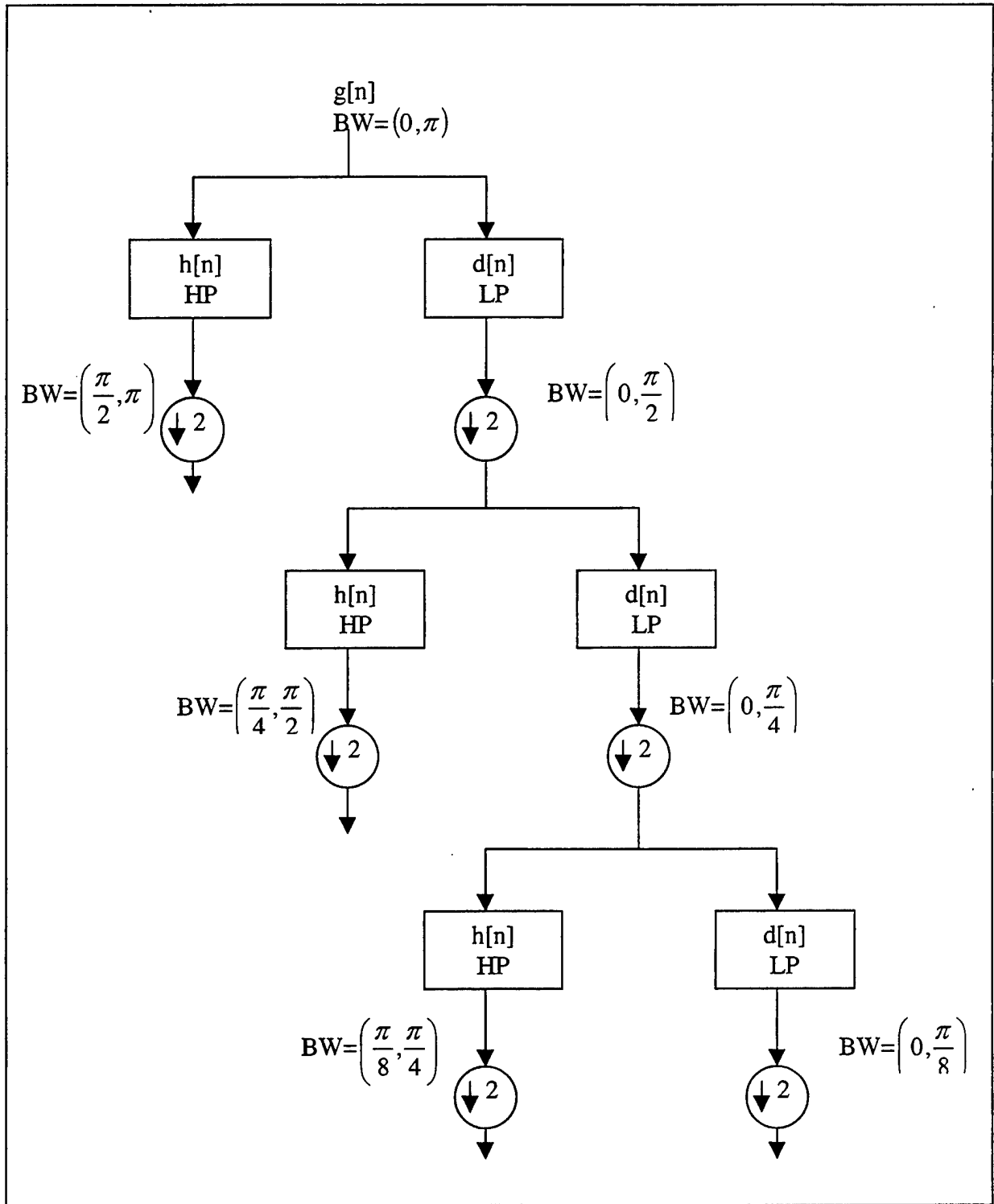
Figure 3.2: The Subband coding algorithm (All bandwidth refers to the original sampling rate).

IV. TIME DIFFERENCE OF ARRIVAL (TDOA) ESTIMATION

In this chapter, we will discuss how to estimate the TDOA between two signals. The TDOA can be employed to find the position of a GSM emitter. Figure 4.1 shows a typical configuration; one emitter and a pair of receivers. The signals are shown in their discrete time form.
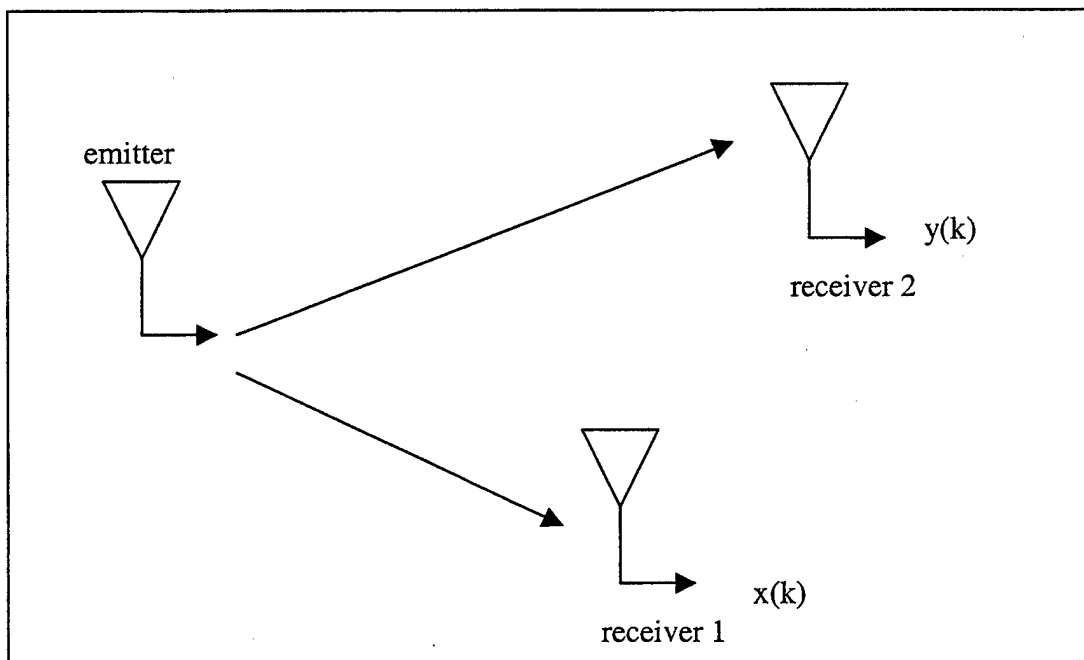
Figure 4.1: One transmitter-two receiver configuration.

The correlation function can be used to estimate the TDOA. In the next section, we will discuss about how to calculate the correlation function and to determine the TDOA from it.

A. CORRELATION FUNCTION

Frequently we would like to know the association between two signals, that is, how one signal is related to the other. Correlation of signals is often encountered in radar and sonar processing, digital communications, and other areas of science and engineering.

To be specific, let us suppose that we have situation, as shown in Figure 4.1. The signal at receiver one is denoted by $x(k)$, while $y(k)$ represents the time shifted version of $x(k)$ at receiver 2. With additive noise, $x(k)$ and $y(k)$ can be modeled as:

$$x(k) = s(k) + n_1(k) \tag{4.1}$$

$$y(k) = \alpha s(k - D) + n_2(k) \qquad k = 0, 1, \cdots, N - 1 \quad , \tag{4.2}$$

where $s(k)$ is the unknown source signal, $n_1(k)$ and $n_2(k)$ are additive noises at the receivers, D is the difference in arrival times at the receivers, α is an attenuation coefficient, and N is the number of samples in each snap shot received at the two receivers.

The most widely accepted method for obtaining TDOA (D in Equation 4.2) uses the cross correlation method. Expectation of x(k) and y(k) leads to

$$
\begin{aligned}
R_{xy}(\tau) &= E\{x(k)y(k - \tau)\} \\
&= E\{[s(k) + n_1(k)][\alpha s(k - D - \tau) + n_2(k - \tau)]\} \\
&= E\{\alpha s(k)s(k - D - \tau) + s(k)n_2(k - \tau) + \alpha s(k - D - \tau)n_1(k) + n_1(k)n_2(k - \tau)\}.
\end{aligned}
$$

Since the noise and signal are independent, and the noise has zero mean, all the cross terms in the expectation equal to zero. Also the noises are independent of each other, so the expectation of the $n_1(k)$ and $n_2(k - \tau)$ is also zero. The theoretical cross correlation of x(k) and y(k) becomes

$$R_{xy}(\tau) = \alpha R_s (D + \tau) \quad . \tag{4.3}$$

Figure 4.2 shows the circuit for cross correlation estimation and the block diagram for the discrete time cross correlator. The cross correlation approach requires that receivers share a precise time reference. The performance of the TDOA estimates can be improved by increasing the summation interval. Once the cross correlation function is computed, the value of τ which maximizes Equation 4.3 is used as the estimate of the TDOA.
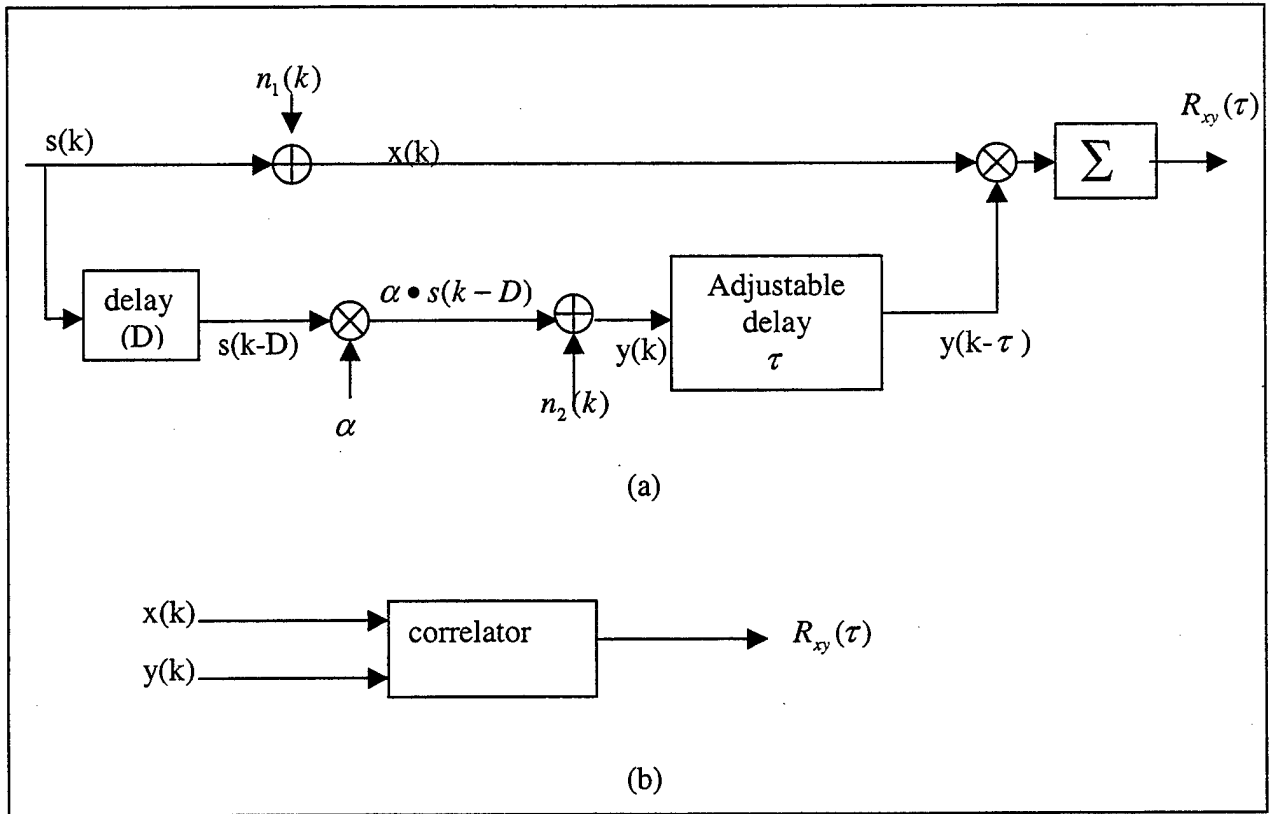
Figure 4.2: (a) Circuit diagram for measuring the cross correlation function of *x(k)* and *y(k)* (b) Block diagram for the discrete time cross correlator.

Figure 4.3 displays the equivalent fast correlation method. A cross spectral density estimate is obtained in the frequency domain, and the cross correlation estimate is obtained via an inverse Fourier transform. The fast correlation method requires appropriate zero padding prior to taking the Fourier transform of x(k) and y(k).

B.   SIMULATION

Simulations were carried out to evaluate the performance of the cross correlation method in performing the TDOA estimate of signals arriving at two separate receivers. The ith error is given by
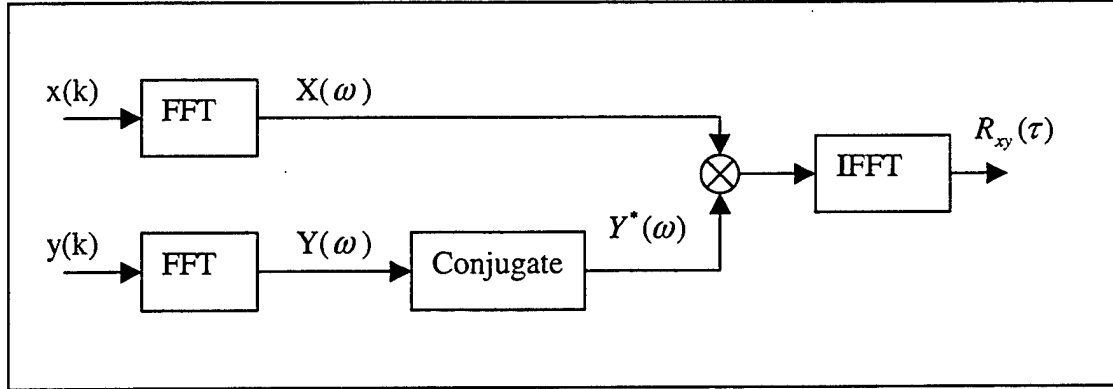
Figure 4.3: Fast cross correlation method using fast Fourier transforms.

$$e_i = D - \hat{D}_i \quad i = 1, 2, \cdots, N \quad , \qquad (4.4)$$

where N is the number of realizations. The mean square error (MSE) is given by

$$MSE = \frac{1}{N} \sum_{i=1}^{N} |e_i|^2 \quad , \qquad (4.5)$$

where e_i is the error for the ith realization. In this part of the experiment the 13-bit Barker code ([+ + + + + - - + + - + - +]) was used, because it has good correlation properties. A

32

200 noise realizations were used and the delay D for this part was fixed at a value of 25 time samples. We simulated two modulation scenarios.

i. The symbol + was modulated by a sinusoid of one period over 32 time samples (the sampling frequency fs=1/32) while the symbol – phase shifted the sinusoid by 180 degree relative to the + symbol.

ii. The symbol + was modulated by a sinusoid of one period over 8 time samples (the sampling frequency fs=1/8) while the symbol – phase shifted the sinusoid by 180 degree relative to the + symbol.

Figure 4.4 plots the mean square error versus to the signal-to-noise ratio (SNR) for case (i) and case (ii). The mean square error depends on the sampling rate. We note that, as the number of samples increases relative to the bit period, the mean square error decreases.
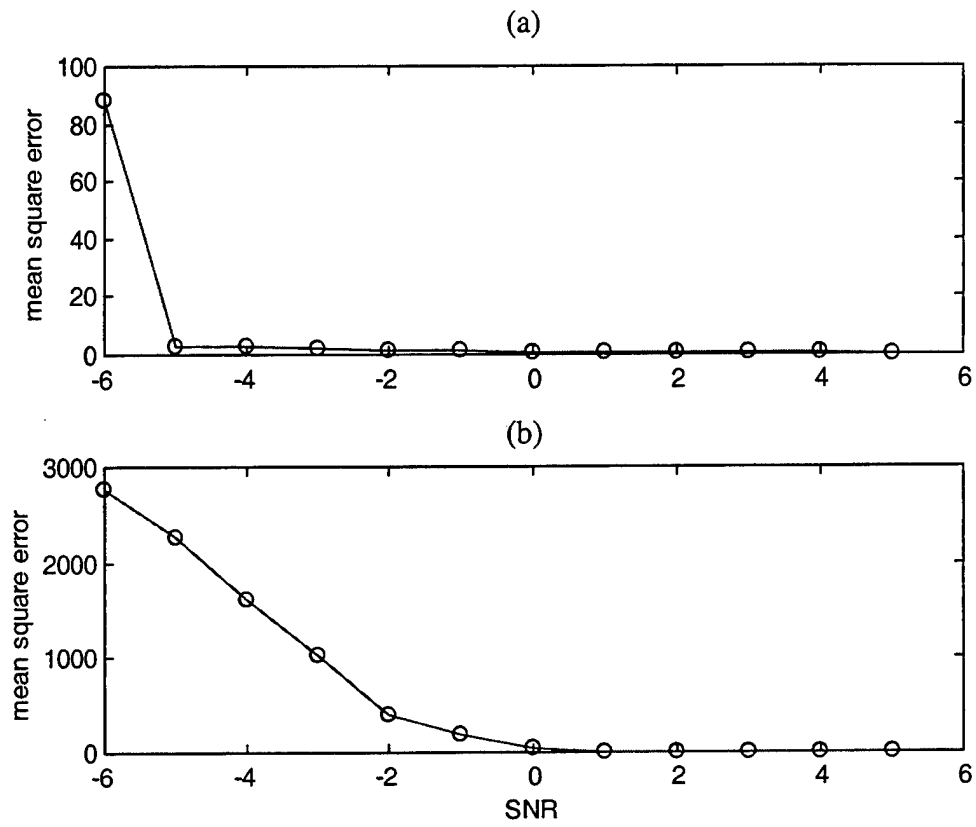
Figure 4.4: (a) The mean square error in estimating the TDOA using cross correlation of time domain signals for case i (b) for case ii.

V. WAVELET DENOISING

Wavelet decompositions have many applications in signal processing. One important one is noise reduction. Each transform coefficient represents a measure of the correlation between the signal and the basis function. Large coefficients represent good correlation, conversely small coefficients represent poor correlation. The main idea in denoising is to retain the coefficients that preserve the signal while removing those that represent noise.

Two main properties of the wavelet transform assist in separating the noise coefficients from the rest. The first property is that, by choosing the basis function to match the signal, the resulting decompositions will contain relatively few coefficients. The second property is that, for a Gaussian noise input, the transform coefficients will remain Gaussian. The orthogonal wavelet transform is a linear operation, which will transform Gaussian noise into Gaussian noise [18].

The wavelet shrinkage algorithm introduced by Donoho and Johnstone [18, 19, 20] keeps only the significant coefficients, representing the signal based on non-linear thresholding. It discards the coefficients that fall below a given magnitude. Wavelet denoising consists of three steps.

1. Taking the wavelet transform of the input signal.

2. Selecting a threshold and suppressing the noisy coefficients by applying a non-linear thresholding technique.

3. Performing the inverse wavelet transform based on the modified coefficients to reconstruct the signal.

A.   CALCULATING A THRESHOLD VALUE

The term threshold refers to a constant that is computed as the cut-off value to separate the coefficients that will be retained from those that will be suppressed. The noisy data can be expressed,

$$x(k) = s(k) + \sigma n(k) \qquad\qquad k = 0, 1, 2, \cdots, N \quad, \qquad\qquad (5.1)$$

where $s(k)$ is the input signal to be recovered from the noisy data, $n(k)$ is zero mean, unit variance additive white Gaussian noise (AWGN), N is the length of the signal, σ is the standard deviation of the AWGN. Since the wavelet transform is a linear operator, the wavelet coefficients will be in the form of Equation 5.1

Algorithms computing the threshold value T require estimation of σ. Five methods of computing thresholds are described below.

1.   Stein's Unbiased Risk Estimator (SURE) Threshold (T_{SU})

This method of threshold calculation was proposed by Donoho and Johnstone [20]. The thresholding is adaptive: A threshold level is assigned to each dyadic resolution level by the principle of minimizing the Stein Unbiased Estimate of Risk (SURE) [21] for threshold estimates. The SURE threshold is smoothness adaptive. If the unknown signal contains jumps, the reconstruction does also. If the unknown signal has a smooth segment, the reconstruction is as smooth as the mother wavelet will allow. This statistical procedure calculates the estimated mean square error (risk) for a range of threshold values, and selects T_{SU} with the resulting minimum risk.

2.   Sqtwolog Threshold (T_{sq})

Sqtwolog threshold uses a fixed form threshold yielding minimax performance multiplied by a small factor proportional to *log(length(signal))* [7]

36

$$T_{sq} = \sqrt{2\log(length(signal))} \qquad . \qquad\qquad (5.2)$$

3. Heursure Threshold (T_h)

Heursure threshold is mixture of SURE and sqtwolog threshold methods [7].

4. Minimaxi Threshold (T_m)

Minimaxi threshold uses a fixed threshold chosen to yield mimimax performance for mean square error against an ideal procedure [7]. The minimax principle is used in statistics in order to design estimators. It is designed to select estimators that minimize the worst case error occurring in the set.

5. Wo-So-Ching Threshold (T_{wsc})

This threshold method was proposed by Wo, So, and Ching. It will be discussed in Chapter VI.

The threshold techniques, 1 through 4, are part of Donoho's method. Details are discussed in Chapters VI and VII. Donoho's method used at the low SNR's did not result in improvement. Threshold technique 5 is used successfully and details are discussed in Chapter VI and VII.

B. THRESHOLDING METHODS (SHRINKAGE)

Once a threshold value is established, a number of methods exist to apply the threshold to suppress or modify the coefficients of decomposition. We examined three thresholding methods.

1. Hyperbolic Thresholding

Hyperbolic thresholding was proposed by Wong and will be discussed in Chapter VI.

2.   Hard Thresholding

Hard thresholding can be described as the usual process of setting to zero the elements whose absolute values are lower than the threshold [7]. For notational convenience, we define $x(k)$ and $n(k)$ as vectors.

$$X = [x(0), x(1), x(2), \cdots, x(N-1)]^T$$
$$N = [n(0), n(1), n(2), \cdots, n(N-1)]^T \qquad (5.3)$$

Let W be NxN a wavelet transform matrix. In a vector form, the transformed output C is related to the input vector X by $C = W X$, where

$$C = [c(j,i), \ j = -1,0,1,\cdots,J; \ i = 0,1,2,\cdots,2^{J-1}] \ , \quad (5.4)$$

and $J = \log_2(N)$. The indices j and i represent the scale and the position in each scale, while c(-1,0) denotes the remaining low-pass filtered coefficients.

The non-linear hard threshold can be defined

$$\hat{c}(i,j) = \begin{cases} c(i,j) & ; \ for \ |c(i,j)| \geq T \\ 0 & ; \qquad otherwise \ . \end{cases} \qquad (5.5)$$

As seen in Figure 5.1, hard thresholding of the transformed coefficients retains the coefficients that exceed the threshold value, while all others are set to zero.

3.   Soft Thresholding

As we see in Figure 5.2, soft thresholding is an extension of hard thresholding. It first sets to zero the elements whose absolute value is lower than the threshold, and then shrinks the remaining non-zero coefficients by the threshold amount [7].

The non-linear soft thresholding can be defined

$$\hat{c}(i,j) = \begin{cases} sign(c(i,j))[|c(i,j)| - T] & ; \ for \qquad |c(i,j)| \geq T \\ 0 & ; \qquad otherwise \ . \end{cases} \qquad (5.6)$$

38

The advantage of this method is that the results are not as sensitive to the precise value of the threshold T, as in the "keep or kill" strategy of the hard thresholding. The disadvantage of this method is that the general shape of the signal might be slightly affected since the even the large coefficients are modified when using this scheme.

The thresholding techniques 2 and 3 are part of Donoho's method, which did not lead to improvement. Thresholding method 1, *hyperbolic thresholding*, was used exclusive in our work. Details can be found in Chapter VI and VII.
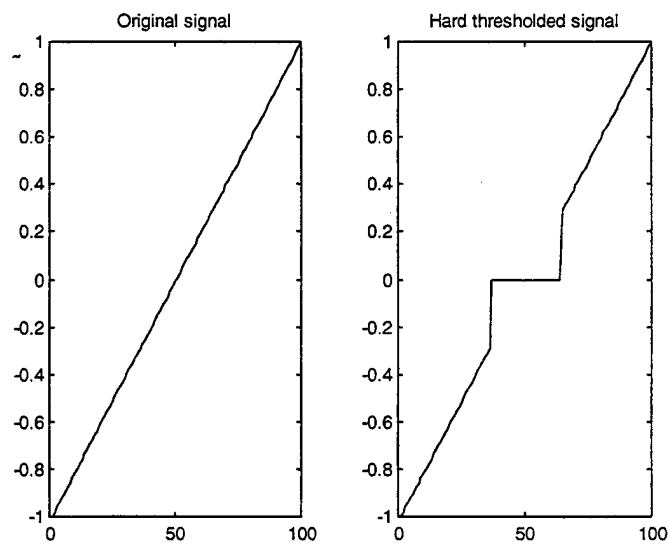
Figure 5.1: Hard thresholding applied to a data segment.
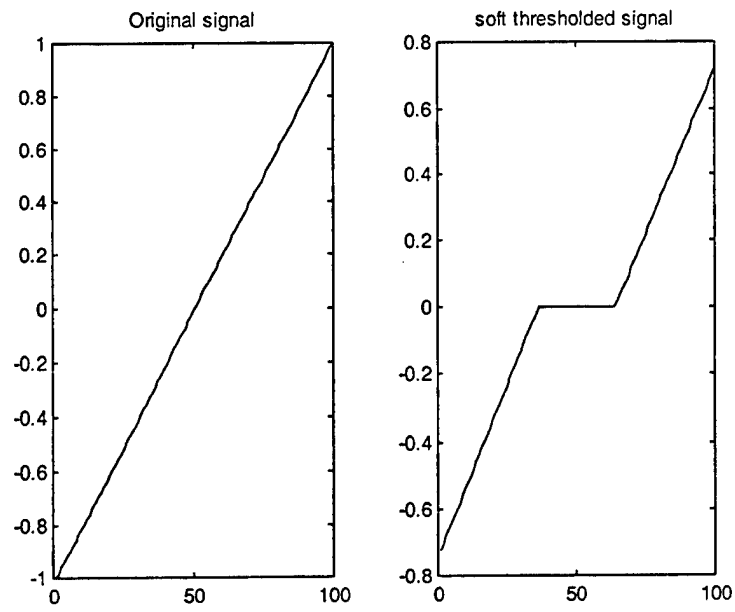
Figure 5.2: Soft thresholding applied to a data segment.

VI. IMPROVED TDOA ESTIMATION

In this chapter, we will discuss the performance of several denoising schemes designed to improve TDOA estimation. Seven methods are presented and evaluated. These tend to reduce the effective noise at the receivers. The methods are: wavelet denoising based on Donoho's method [7], wavelet denoising using the Wo-So-Ching threshold. [15], wavelet denoising using hyperbolic shrinkage [16], wavelet denoising using median filtering [14], a modified approximate maximum-likelihood delay estimation based in part on [17], denoising based on the fourth order moment, and a time varying technique. Figure 6.1 is a generic figure for all methods addressed in this chapter.

A.   WAVELET DENOISING BASED ON DONOHO'S METHOD

The wavelet denoising as proposed by Donoho [7, 18, 19, and 20], was discussed in Chapter V. This method fails at low SNR's and the results can be found in Chapter VII.

B.   WAVELET DENOISING USING THE WO-SO-CHING THRESHOLD

Wavelet denoising is applied to the noisy signals, which are obtained at the two spatially separated sensors. Prior to cross correlation, each one of the sensor outputs is denoised according to the Wo-So-Ching thresholding rule to increase the effective input signal-to-noise ratio (SNR). The proposed system consists of two sub-units (Wavelet denoising and Correlator) as depicted in Figure 6.1.

Wavelet denoising (WD) is applied to each received signal to recover the corresponding source waveform. The restored signals are cross correlated. The TDOA

estimate is given by the argument at which the cross correlation function attains its maximum value.
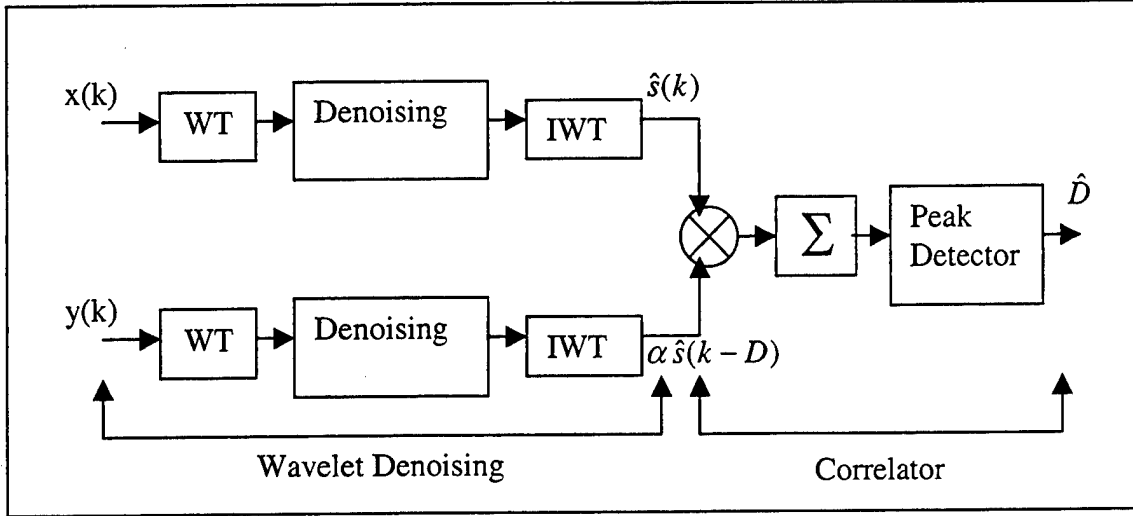
Figure 6.1: System block diagram for TDOA estimate using wavelet denoising.

The wavelet denoising technique consists of three steps; taking the wavelet transform of the received signals, thresholding to modify the wavelet coefficients, and performing the inverse wavelet transform of the modified coefficients. For notational convenience, we define the $x(k), y(k), s(k), n_1(k), and\ n_2(k)$ sequences in vector form as

$$X = [x(0),\ x(1),\ x(2),\ \cdots,\ x(N-1)]^T$$
$$Y = [y(0),\ y(1),\ y(2),\ \cdots,\ y(N-1)]^T$$
$$S = [s(0),\ s(1),\ s(2),\ \cdots,\ s(N-1)]^T \qquad (6.1)$$
$$N_1 = [n_1(0),\ n_1(1),\ n_1(2),\ \cdots,\ n_1(N-1)]^T$$
$$N_2 = [n_2(0),\ n_2(1),\ n_2(2),\ \cdots,\ n_2(N-1)]^T.$$

The start time "0" denotes the first data point in a given block of data. Let W be a NxN orthonormal wavelet transform matrix. The transformed output C is related to the input

42

vector X by $C = W \, X$. Since W is a linear transform matrix, C can be decomposed into $C = C_S + C_{N_k}$, where

$$C_S = [c_s(j,i), \quad j = -1,0,1,\cdots,J; \quad i = 0,1,2,\cdots,2^{J-1}] \tag{6.2}$$

$$C_{N_k} = [c_{n_k}(j,i), \quad j = -1,0,1,\cdots,J; i = 0,1,2,\cdots,2^{J-1}] \quad , \tag{6.3}$$

are the wavelet transform coefficients of the source signal vector S and the noise vector N_k (k=1, 2) respectively. The indices i and j represent the scale and the position in each scale, respectively, while $c_s(-1,0)$ and $c_{n_k}(-1,0)$ denote the low-pass filtered coefficients.

Note that, C_{N_k} is still a Gaussian vector. The main idea of the signal restoration using wavelet denoising is to adapt each $c(j,i)$ to make its value close to $c_S(j,i)$ so that a good approximation of $s(k)$ can be obtained after taking the inverse wavelet transform.

The Wo-So-Ching threshold is derived according to the Neyman-Pearson criterion as it is used in hypothesis testing [15]. This criterion is stated as follows:

Let c be a Gaussian random variable with known variance σ^2. Let a test be conducted with the following hypotheses

$$H_0 : E\{c\} = \mu_0 \tag{6.4}$$

and,

$$H_1 : E\{c\} \neq \mu_0 \tag{6.5}$$

We will denote the acceptance of hypothesis H_0 and H_1 as D_0 and D_1, respectively. The type II error $P(D_0 / H_1)$ will be minimized for a given $P(D_1 / H_0)$ when the threshold λ is selected as follows

$$|c - \mu_0| \leq \lambda = \sqrt{2}\sigma.erf^{-1}(1-\alpha) \quad , \tag{6.6}$$

where

c is the Gaussian random variable with variance σ^2,

μ_0 is the expectation of c under hypothesis H_0,

λ is the Wo-So-Ching threshold,

α is the type I error that is $\alpha = P(D_1 / H_0)$ and

$$erf(v) = \frac{2}{\sqrt{\pi}} \int_0^v e^{t^2} dt \qquad .$$

$$(6.7)$$

This denoising method discards the individual elements in C_S that are too small in magnitude. The wavelet coefficient $c(j,i)$ is regarded as totally due to noise if

$$|c(j,i)| \leq \lambda = \sqrt{2}\sigma \, erf^{-1}(1-\alpha) \qquad .$$

$$(6.8)$$

As a result, $c_s(j,i)$ will be modified and is given by

$$\hat{c}_s(j,i) = \begin{cases} c(j,i) & , \quad if \ |c(j,i)| \geq \lambda \\ 0 & , \quad otherwise \end{cases}$$

$$(6.9)$$

The denoising process is applied to both $x(k)$ and $y(k)$. The restored signals of $s(k)$ and $\alpha \, s(k-D)$ are denoted by $\hat{s}(k)$ and $\alpha \, \hat{s}(k-D)$, respectively. The TDOA estimate is given by the location of peak of the cross correlation function of modified $x(k)$ and $y(k)$ coefficients.

C. WAVELET DENOISING USING HYPERBOLIC SHRINKAGE

As mentioned earlier, denoising of signals corrupted by white noise uses three steps; the wavelet transform, shrinkage, and the inverse wavelet transform. Shrinkage reduces the value of each coefficient in magnitude by an amount related to the threshold value. The amount of the shrinkage is determined by the shrinkage method. In this section the hyperbolic method is used.

Hyperbolic shrinkage [16] is defined as:

$$\hat{c}(i,j) = \begin{cases} sign[c(i,j)]\sqrt{c(i,j)^2 - \lambda^2} & ; & |c(i,j)| > \lambda \\ 0 & ; & |c(i,j)| \le \lambda \; ; \end{cases} \quad (6.10)$$

where $c(i,j)$ is the detail function of received signal, λ is the Wo-So-Ching threshold (as given in Equation 6.8), and $\hat{c}(i,j)$ is the modified detail function after shrinkage. The coefficients are set to the square root of the difference of the squares of the values as long as the absolute value of the coefficient is greater than the threshold. If the absolute value of the coefficient is less than the threshold then the coefficient is set to zero. Hyperbolic shrinkage can give good performance for a wide variety of signals. Other shrinkage techniques may offer better performance for specific input signals and noise conditions. The proposed denosing system is shown in Figure 6.1

D.   WAVELET DENOISING USING THE MEDIAN FILTER

In this method, a median filter (filter length 3) is applied to the first detail function of the wavelet transforms. The signals are reconstructed using the modified wavelet coefficients. The denoising technique is depicted in Figure 6.1.

The median filter is a one dimensional filtering technique. It is a non-linear technique that applies a sliding window to a sequence [14]. The median filter replaces the center point of the window with the median value of all the points contained in the window. The length of the window is very important. For example, for a stationary signal such as a sinusoid, a longer window length is better. But if the signal is a non-stationary, a short window length is better. This is an important drawback when using the median filter if one does not have a priori information about the source signal.

45

E.   MODIFIED   APPROXIMATE   MAXIMUM-LIKELIHOOD   DELAY   ESTIMATION

An approximate maximum likelihood (AML) algorithm is proposed by Y.T. Chan, H.C. So, and P.C. Ching to estimate the TDOA of signals [17]. The general idea in this method is to attenuate the frequency bands where the noise is strong and to enhance the frequency bands where the signal is strong. To do this, we can use the pre-filters as shown in Figure 6.2. The pre-filters $H_1(f)$ and $H_2(f)$ are chosen as follows [17]

$$H_1(f)H_2^*(f) = \frac{\dfrac{G_{ss}(f)}{G_{n_1n_1}(f)G_{n_2n_2}(f)}}{1 + \dfrac{G_{ss}(f)}{G_{n_1n_1}(f)} + \dfrac{G_{ss}(f)}{G_{n_2n_2}(f)}} \quad . \qquad (6.11)$$

Where $G_{ss}(f)$, $G_{n_1n_1}(f)$, and $G_{n_2n_2}(f)$ denote the auto-power spectra of s(k), $n_1(k)$ and $n_2(k)$ respectively. This choice of $H_1(f)H_2^*(f)$ is known as maximum likelihood (ML) weighting.

When we multiply the denominator and nominator in Equation 6.11 by $G_{n_1n_1}(f)G_{n_2n_2}(f)$ and integrate the power spectral densities, we obtain the formula for the weight for each subband. This tends to enhance frequency bands where the signal is strong.

The weights w_{d_i} are given by

$$w_{d_i} = \frac{\hat{\sigma}_{sd_i}^2}{\hat{\sigma}_{n_1d_i}^2 \hat{\sigma}_{n_2d_i}^2 + \hat{\sigma}_{sd_i}^2 (\hat{\sigma}_{n_1d_i}^2 + \hat{\sigma}_{n_2d_i}^2)} \quad , i = 1, 2, \cdots, J \quad , \qquad (6.12)$$

and are used as indicated in Figure 6.3.

46
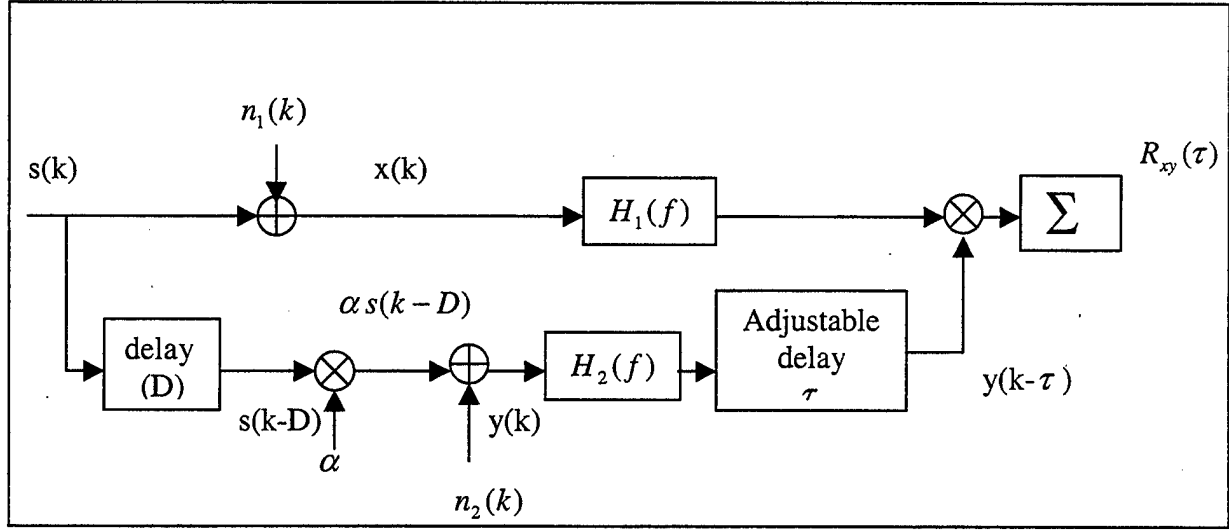
Figure 6.2: A generalized cross correlator.

Since the noise density is flat and the filter gain of the wavelet transform is two, the noise power will essentially remain same after passing through each high and low pass filter. The noise power at each subband can be estimated using the formulas below

$$\hat{\sigma}_{n_1}^2 = \hat{R}_{xx}(0) - \hat{\sigma}_s^2 \qquad (6.13)$$

$$\hat{\sigma}_{n_2}^2 = \hat{R}_{yy}(0) - \hat{\sigma}_s^2 \qquad (6.14)$$

$$\hat{\sigma}_s^2 = \arg\max \hat{R}_{xy}(\tau) \qquad (6.15)$$

$$\hat{\sigma}_{n_1 d_i}^2 = \hat{\sigma}_{n_1}^2 \qquad (6.16)$$

$$\hat{\sigma}_{n_2 d_i}^2 = \hat{\sigma}_{n_2}^2 \qquad . \qquad (6.17)$$

The variance $\sigma_{n_1}^2$, $\sigma_{n_2}^2$ are the noise powers at each receiver, while σ_s^2 is the signal power

at receiver, $\sigma_{n_1 d_i}^2$, $\sigma_{n_2 d_i}^2$ represent the noise powers at the ith subband after the wavelet

transform. The signal power at each subband "i" can be estimated by

$$\hat{\sigma}_{sd_i}^2 = \left(\frac{1}{N} \sum_{k=0}^{N-1} d_i^2(k) \right) - \hat{\sigma}_{n_1 d_i}^2 \ , \ i = 1, 2, \cdots, J \ ; \tag{6.18}$$

where N is the length of detail function d_i, and "i" denotes the scale.

The denoising procedure, shown in Figure 6.3, consists of the three steps. These

steps are discrete wavelet decomposition, scaling of each subband sequence, and an

inverse wavelet transform. The modified subband sequences are obtained by weighting

each subband by

$$d_i^c = w_{d_i} d_i \ ; \ i = 1, 2, \cdots, J \ . \tag{6.19}$$

The modified AML method is applied to both channels. The modified subband

sequences $d_1^c, d_2^c, \cdots, d_L^c$ and a_L^c are then combined to form the modified AML denoised

signals. The MATLAB codes for modified approximate maximum-likelihood technique

are presented in the Appendix.

F.   WAVELET DENOISING BASED ON THE FOURTH ORDER MOMENT

1.   White Noise

White noise is the term applied to any zero mean random process whose power

spectral density is constant. Since the power spectral density function is constant, the

correlation function for white noise is an impulse. In other words, the samples of white

noise process are uncorrelated. The correlation function of white noise process $u[n]$ has

the form

48

$$R_u[n_1, n_0] = \sigma_0[n]\delta[n_1 - n_0] \quad . \tag{6.20}$$

Usually a white noise process is taken to a mean a stationary white noise process so that

$$R_u[l] = \sigma_0^2 \delta[l] \tag{6.21}$$

and

$$S_u(e^{j\omega}) = \sigma_0^2 \quad . \tag{6.22}$$

If u is a random vector of N consecutive samples from white Gaussian process, the probability density function is given by

$$f(u_0, u_1, \cdots, u_{N-1}) = \frac{1}{(2\pi\sigma_0^2)^{N/2}} e^{-\frac{\|u\|^2}{2\sigma_0^2}} = \prod_{n=0}^{N-1} \frac{1}{\sqrt{2\pi\sigma_0^2}} e^{-\frac{(u_n)^2}{2\sigma_0^2}} \quad . \tag{6.23}$$

The samples of a Gaussian white noise process are uncorrelated and hence independent.
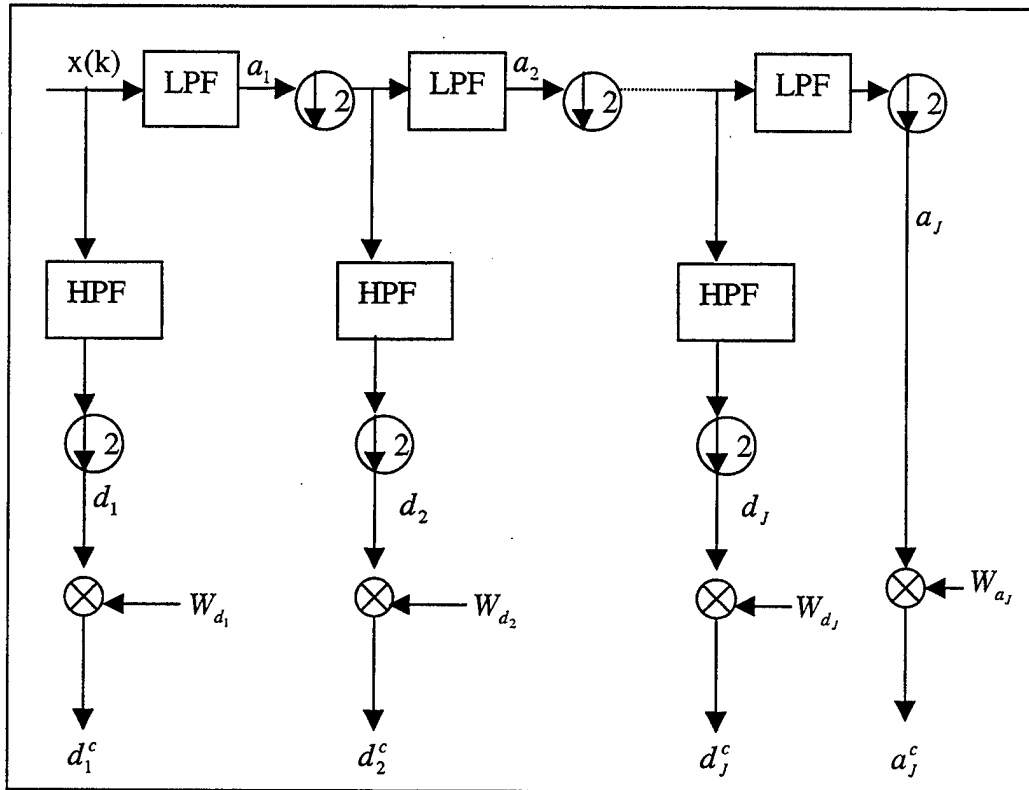
Figure 6.3: Block diagram for scaling of each subband.

2. Moments of White Noise Process

The definitions of moments for a stationary random process are;

$$M_u^{(1)} = E\{u[n]\} = m_u \tag{6.24}$$

$$M_u^{(2)}[l_1] = E\{u[n]u[n+l_1]\} = R_u[l_1] \tag{6.25}$$

$$M_u^{(3)}[l_1,l_2] = E\{u[n]u[n+l_1]u[n+l_2]\} \tag{6.26}$$

$$M_u^{(4)}[l_1,l_2,l_3] = E\{u[n]u[n+l_1]u[n+l_2]u[n+l_3]\} \quad . \tag{6.27}$$

The first two moments are equal to the mean and correlation function defined earlier. The fourth moment is

$$
\begin{aligned}
M_u^{(4)}[l_1,l_2,l_3] &= E\{u[n]u[n+l_1]u[n+l_2]u[n+l_3]\} \\
&= E\{u[n]u[n+l_1]\}E\{u[n+l_2]u[n+l_3]\} \\
&\quad + E\{u[n]u[n+l_2]\}E\{u[n+l_1]u[n+l_3]\} \\
&\quad + E\{u[n]u[n+l_3]\}E\{u[n+l_1]u[n+l_2]\} \quad .
\end{aligned}
\tag{6.28}
$$

If $l_1 = l_2 = l_3 = 0$

$$
\begin{aligned}
M_u^{(4)}[0,0,0] &= E\{u[n]u[n]\}E\{u[n]u[n]\} \\
&\quad + E\{u[n]u[n]\}E\{u[n]u[n]\} \\
&\quad + E\{u[n]u[n]\}E\{u[n]u[n]\} \\
&= 3\sigma_u^4 \quad .
\end{aligned}
\tag{6.29}
$$

3. The Wavelet Transform of White Noise

Since the wavelet transform is a linear operation, the Gaussian process remains Gaussian after passing through low and high pass filters and hence preserves the high order moments properties.

4. Fourth Order Moment of the Received Signal

The received signals are of the form

$$x(k) = s(k) + n_1(k) \tag{6.30}$$

$$y(k) = \alpha s(k-D) + n_2(k) \quad , \qquad (6.31)$$

where $n_1(k)$ and $n_2(k)$ are statistically independent, zero mean Gaussian random variables. The fourth order moment estimation of the received signal is,

$$
\begin{aligned}
M_x^{(4)}[0,0,0] &= E\{x^4[k]\} \\
&= E\{[s[k]+n_1[k]]^4\} \\
&= E\{s^4[k]\} + 4E\{s^3k]n_1[k]\} + 6E\{s^2[k]n_1^2[k]\} \\
&\quad + 4E\{s[k]n_1^3[k]\} + E\{n_1^4[k]\} \\
&\geq 3\sigma_{n1}^4 \quad .
\end{aligned}
\qquad (6.32)
$$

5. Mean and Standard Deviation of the Fourth Order Moment

If we call $z = M_u^{(4)}$, the mean of the z becomes,

$$E\{z\} = \frac{1}{N}\sum_{i=0}^{N-1}E\{u_i^4\} = \frac{1}{N}\sum_{i=0}^{N-1}3\sigma_u^4 = 3\sigma_u^4 \quad . \qquad (6.33)$$

The second moment of the z is,

$$
\begin{aligned}
E\{z^2\} &= E\left\{\frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1}u_i^4 u_j^4\right\} \\
&= \frac{1}{N^2}\sum_{i=0}^{N-1}\overline{u}_i^8 + \frac{N(N-1)}{N^2}\left(\overline{u}_i^4\overline{u}_j^4\right) \\
&= \frac{105\sigma_u^8}{N} + \frac{(N-1)9\sigma_u^8}{N} = \frac{\sigma_u^8}{N}(9N+96).
\end{aligned}
\qquad (6.34)
$$

The variance of the z is,

$$
\begin{aligned}
\sigma_z^2 &= E\{z^2\} - E^2\{z\} \\
&= \frac{\sigma_u^8}{N}(9N+96) - 9\sigma_u^8 \\
&= \frac{96}{N}\sigma_u^8 \quad .
\end{aligned}
\qquad (6.35)
$$

If N is large the, variance is close to zero. We accounted for the variance in choosing a threshold, see Equation 6.42. Experimentally, this amounted to a change from $3\sigma_{n_1}^4$ to $3.1\sigma_{n_1}^4$.

6. Denoising Based on the Fourth Order Moment

Since the wavelet transform is linear, each detail function in Figure 6.3 is in the same form as Equation 6.30 and 6.31. The fourth order moment of detail function which contains signal should be greater than $3\sigma_{nd_i}^2$, where $\sigma_{nd_i}^2$ denotes the noise power at subband d_i. Using this property we can eliminate the wavelet coefficients which represent noise and hold the ones which represent the signal.

We can estimate the noise power using the formulas below

$$\sigma_s^2 = \arg\max_{\tau} R_{xx}(\tau) \tag{6.36}$$

$$\sigma_{n_1}^2 = R_{xx}(0) - \sigma_s^2 \tag{6.37}$$

$$\sigma_{n_2}^2 = R_{yy}(0) - \sigma_s^2 \quad . \tag{6.38}$$

The noise power after passing through the first high pass filter is

$$\sigma_{n_j d_1}^2 = G.\sigma_{n_j}^2 \ /2 \quad j = 1,2 \quad , \tag{6.39}$$

where G is the gain of the high and low pass filters. If we generalize this formula, the noise power of any detail function d_i is

$$\sigma_{n_j d_i}^2 = \frac{G^i \sigma_{n_j}^2}{2^i} \quad j = 1,2 \quad . \tag{6.40}$$

And we assume that filter gain is 2 then

$$\sigma_{n_j d_i}^2 = \sigma_{n_j}^2 \quad j = 1, 2 \quad . \tag{6.41}$$

Using the fourth order moment property we can modify the each detail function in Figure 6.3 as given by

$$d_i^c = \begin{cases} d_i & ; if \ E\{d_i^4\} \geq (3\sigma_{n1}^4 \pm \sigma_z^2) \\ 0 & ; otherwise \end{cases} \qquad (6.42)$$

After modifying all detail functions, we perform the inverse wavelet transform to reconstruct the denoised signal. The MATLAB codes for wavelet denosing based on the fourth order moment technique are presented in the Appendix.

G.   TIME VARYING TECHNIQUE

The modified AML delay estimation method is based on the idea of attenuating the spectral components where the signal is weak. This method works well if the subband information is noise only or signal only for the duration of the segment.

Since some signals are time varying or transient, the subbands could contain signal only for parts of the segment. We modified this method by allowing for some variations over time. In particular, we divided each detail output into two time segments and applied the equations of section E to each segment. This approach allowed a time varying gain (i.e w_{d_i} weights for the first half of the segment, w_{d_j} weights for the second half of the segment). The MATLAB codes for time varying modified approximate maximum-likelihood technique (time varying MAML) are presented in the Appendix.

53

VII. SIGNAL DESCRIPTION AND SIMULATION RESULTS

A. SIGNAL DESCRIPTION

We used two different types of test signals, a generic signal and a GSM signal. These two signals are described in this section.

1. Generic Signals

In this part of the simulation we used the 13 bit Barker code sequence as the message to create four different test signals. The generic signal representing one code bit is of the form $\sin(2*\pi*f*n/N)$ where $N=32$, $n=0, 1, ..., 31$. Signals A, B, C, and D use value of f of 1, 4, 8, and 12 respectively. The generic signals are plotted in Figure 7.1. The MATLAB codes for generic signal generation are presented in Appendix.
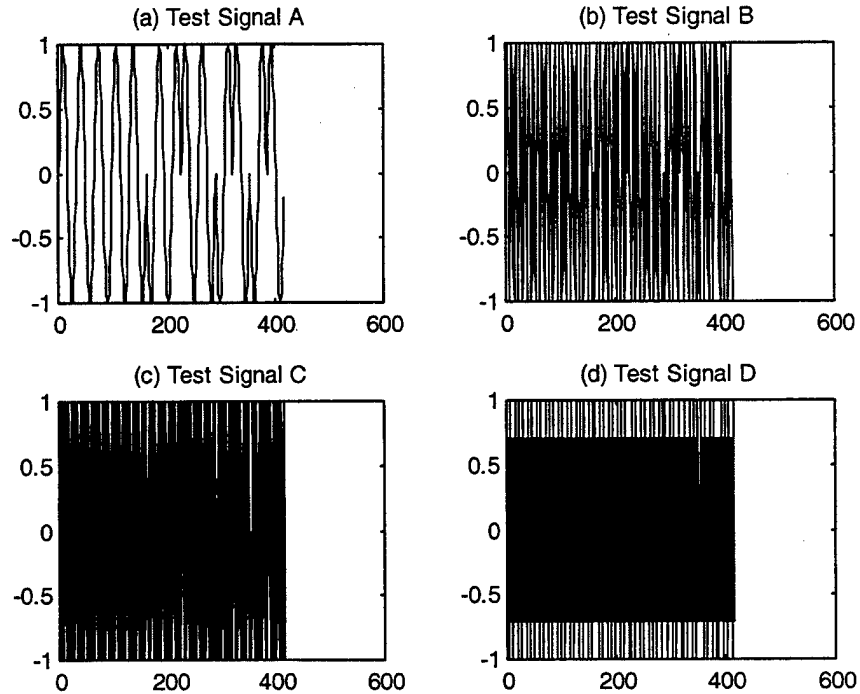
Figure 7.1: (a) Generic signal A, (b) Generic Signal B, (c) Generic signal C, (d) Generic signal D.

2. GSM Signal

The GSM signal properties are detailed in Chapter II. According to these properties we simulated the GSM signal. The I and Q channel of the GSM base band signal are shown in Figure 7.2. The MATLAB codes for GSM signal generation are presented in Appendix [4].
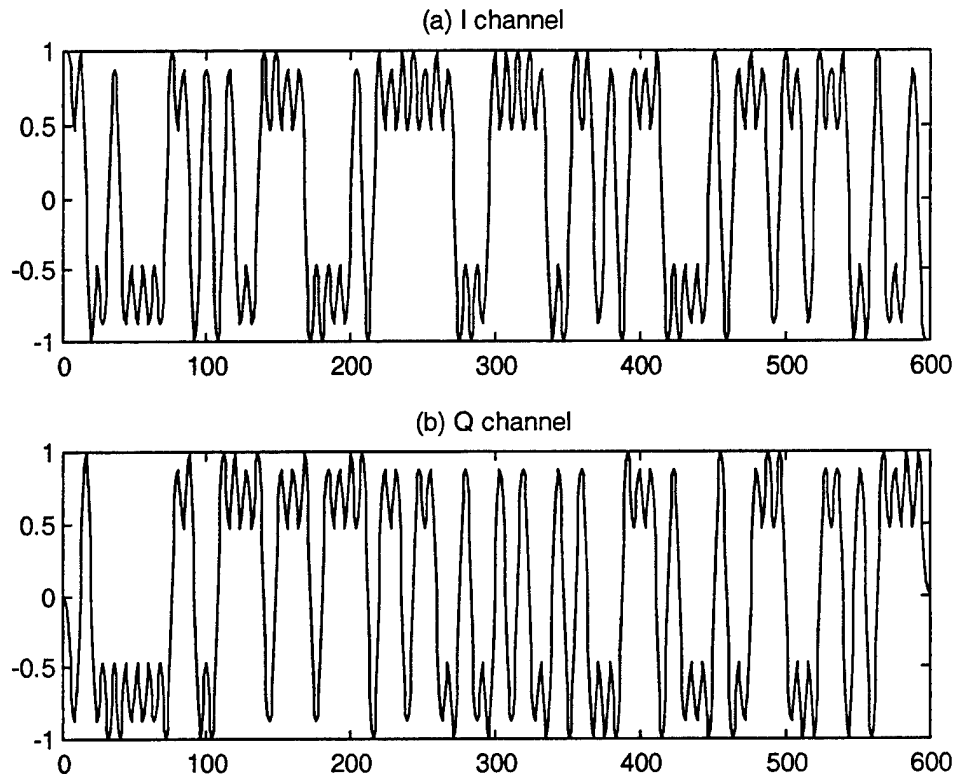
Figure 7.2: (a) I channel of GSM signal (b) Q channel of GSM signal.

3.   Delay Definition

For the generic signal simulations the delay was randomly chosen between −30 to +30. For the GSM signal simulations the delay was randomly chosen between −150 to +150. The realizations of delays are shown in Figure 7.3 in a histogram fashion. Both randomizations used the same seed number.
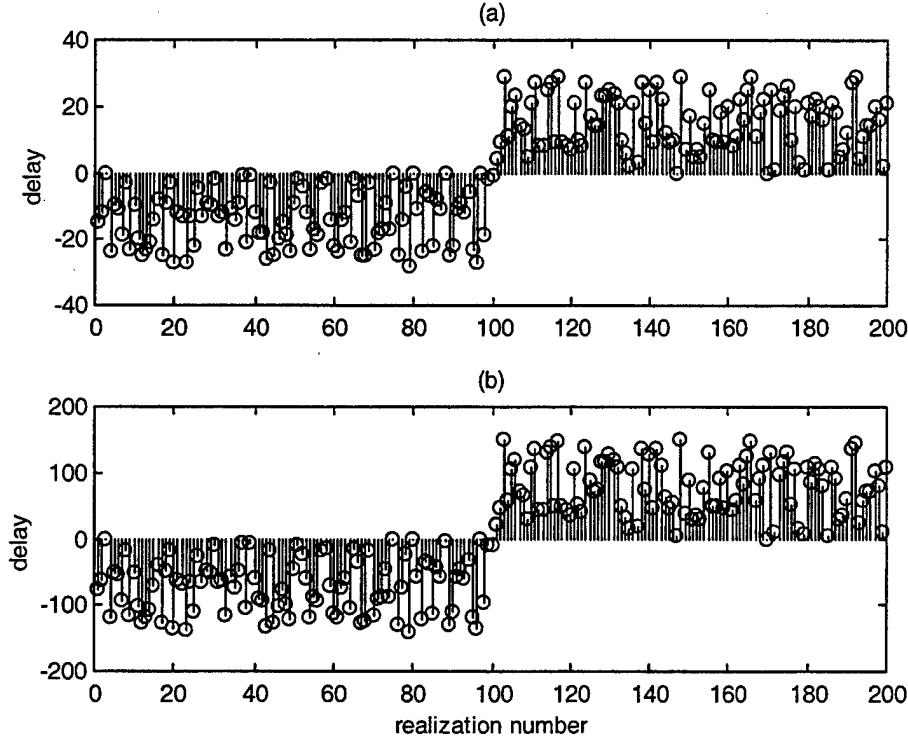
Figure 7.3: (a) Delay for generic signals, (b) Delay for the GSM signals.

B.   SIMULATION RESULTS

We used the mean square error (MSE) as the criteria of goodness (i.e. low MSE denotes a small error and hence good localization). The MSE is defined in Equation 4.5. To quantify the global improvement, we compute the sum of the MSE's obtained from a given method and compare it to the total error for the non-denoised TDOA estimation.

The improvement is given in percentage. The total error is obtained by summing the MSE over all SNR's of interest (i.e., -3 to +5 dB).

1. Donoho's Method

Donoho proposed different types of thresholds and thresholding techniques. This is explained in Chapter V. All of this methods were tried on the generic signals. The mean square error using wavelet denoising based on Donoho's method did not improve relative to the non-denoised version at the low SNR values of interest. The mean square error obtained from the correlation of the time domain raw signals is much smaller than the ones obtained using Donoho's method.

2. Wavelet Denoising Using the Wo-So-Ching Threshold

a. Simulation Results for the Generic Signals

As seen in Figure 7.4 (a), there is an slight improvement in the mean square error using Wo-So-Ching threshold method for generic signal A. As the carrier frequency is increased (Figure 7.4 (b) and Figure 7.5), the mean square error also increase. As long as the number of samples per binary bit is high there is an improvement using this denoising method, but the advantage disappears as the carrier frequency increases. The results are summarized in Table 7.1.

b. Simulation Results for the GSM Signals

As seen in Figure 7.6, there is about 28% improvement in the total mean square error using Wo-So-Ching thresholding method for GSM signal.

Figure 7.4: (a) MSE versus SNR with and without denoising using the Wo-So-Ching thresholding method for generic signal A. (b) MSE for generic signal B.

Figure 7.5: (a) MSE versus SNR with and without denoising using the Wo-So-Ching threshold method for generic signal C. (b) MSE for generic signal D.

SNR	Xcorr (A)	Wo-So-Ching threshold (A)	Xcorr (B)	Wo-So-Ching threshold (B)
5	0.385	0.25	0	0
4	0.415	0.275	0	0
3	0.485	0.425	0	0
2	0.715	0.535	0	0
1	0.795	0.78	0.32	0.32
0	0.805	0.725	0	0.96
-1	1.145	1.02	2.89	3.84
-2	1.62	1.34	4.505	6.49
-3	2.085	1.815	8.38	12.36

Table 7.1: (a) MSE versus SNR for with and without denoising using the Wo-So-Ching threshold method for generic signal A and B.

SNR	Xcorr (C)	Wo-So-Ching threshold (C)	Xcorr (D)	Wo-So-Ching threshold (D)
5	0	0	0	0
4	0	0.16	0	0
3	0	0.56	0	0
2	0.08	0.64	0	0
1	1.2	3.52	0	0
0	2.72	5.84	0.32	1.92
-1	3.52	8.56	1.6	5.12
-2	5.92	16.56	5.095	9.7
-3	8.88	28.08	5.665	15.62

Table 7.1: (b) MSE versus SNR for with and without denoising using the Wo-So-Ching threshold method for generic signal C and D.
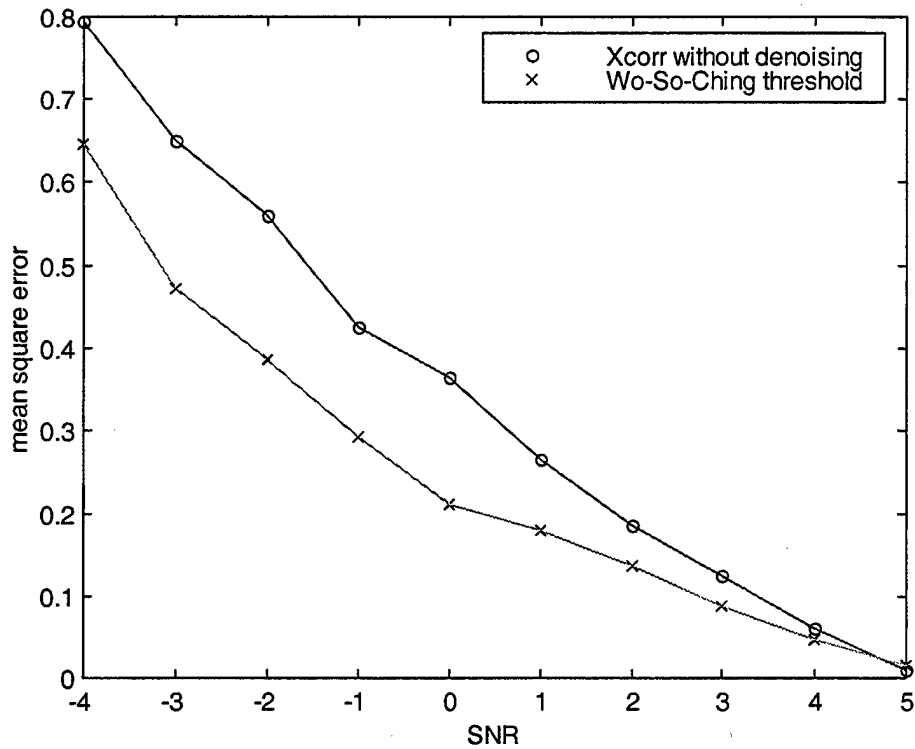
Figure 7.6: MSE versus SNR with and without denoising using the Wo-So-Ching threshold method for the GSM signal.

3. Wavelet Denoising Using the Hyperbolic Shrinkage

a. Simulation Results for the Generic Signals

As seen in Figure 7.7, there is an improvement in the mean square error using hyperbolic shrinkage method for generic signal A and B. Again as the carrier frequency is increased (Figure 7.8), the mean square error also increase. The results are summarized in Table 7.2. If we compare the Table 7.1 and Table 7.2, we can conclude that the hyperbolic shrinkage method gives better result than the Wo-So-Ching threshold method.

b. Simulation Results for GSM Signals

As seen in Figure 7.9, there is about 48% improvement in the total mean square error using Hyperbolic Shrinkage method for GSM signal. If we compare the Figure 7.6 and 7.9 we can conclude that hyperbolic shrinkage method gives better result for GSM signal than Wo-So-Ching thresholding method.

Figure 7.7: (a) MSE versus SNR with and without denoising using the Hyperbolic Shrinkage method for generic signal A. (b) MSE for generic signal B.

Figure 7.8: (a) MSE versus SNR with and without denoising using the Hyperbolic

Shrinkage method for generic signal C. (b) MSE for generic signal D.

SNR	Xcorr (A)	Hyperbolic shrinkage (A)	Xcorr (B)	Hyperbolic shrinkage (B)
5	0.375	0.2000	0	0
4	0.53	0.3200	0	0.08
3	0.52	0.4100	0	0.08
2	0.625	0.5400	0	0.16
1	0.96	0.7350	0.32	0.24
0	1.07	0.8200	0	1.12
-1	1.345	1.0050	2.89	2.8
-2	1.36	1.2100	4.505	3.6
-3	1.895	1.3300	8.38	7.84
-4	2.425	2.2400	12.025	10.465
-5	2.57	3.5350	38.24	17.185
-6	87.41	14.5850	268.01	190.41

Table 7.2: (a) MSE versus SNR for with and without denoising using Hyperbolic

shrinkage method for generic signals A and B.

63

SNR	Xcorr (C)	Hyperbolic shrinkage (C)	Xcorr (D)	Hyperbolic shrinkage (D)
5	0	0.1200	0	0
4	0	0.3200	0	0.24
3	0.08	0.4800	0	0.32
2	0.16	1.0000	0	1.76
1	1.04	2.0400	0	2.8
0	1.6	3.2200	0	4.015
-1	4.08	4.3800	1.645	7.625
-2	4.08	5.0800	0.96	12.735
-3	9.76	10.6450	8.76	87.35
-4	14.8	16.6750	15.385	29.055
-5	24.96	20.1500	22.97	680.8
-6	118.8	163.7250	57.635	1278.2

Table 7.2: (b) MSE versus SNR for with and without denoising using Hyperbolic shrinkage method for generic signals C and D.
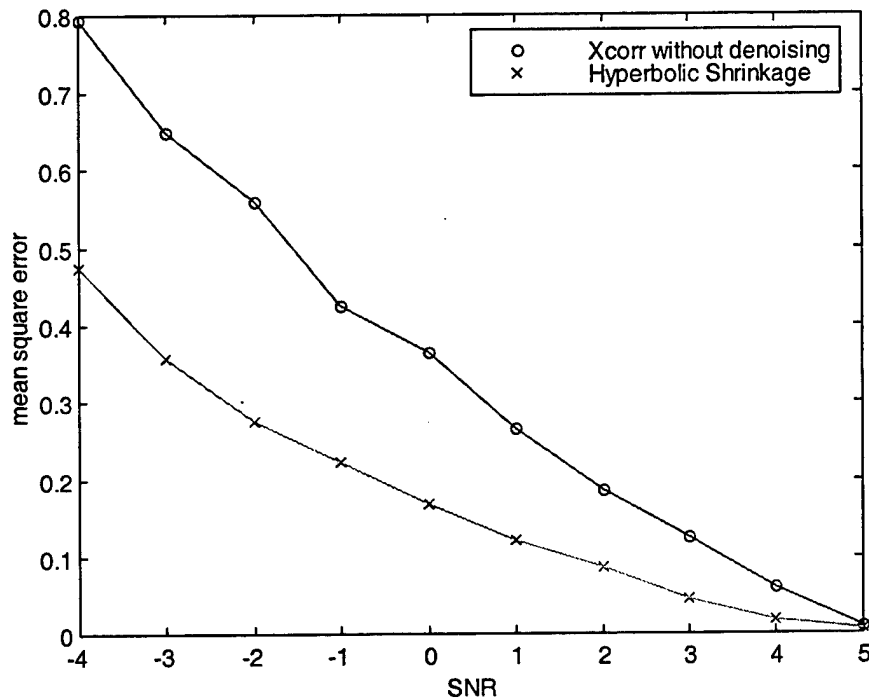
Figure 7.9: MSE versus SNR with and without denoising using the Hyperbolic Shrinkage method for the GSM signal.

4.   Wavelet Denoising Using the Median Filter

a.   Simulation Results for the Generic Signals

As seen in Figure 7.10, there is an improvement in the mean square error using median filtering on generic signals A and B. Again as the carrier frequency is increased (Figure 7.11), the mean square error also increase. The results are summarized in Table 7.3. If we compare the Table 7.1, 7.2 and 7.3, we can conclude that median filtering has the best result for the generic signal B but for the other signals the Hyperbolic shrinkage method provides better results.
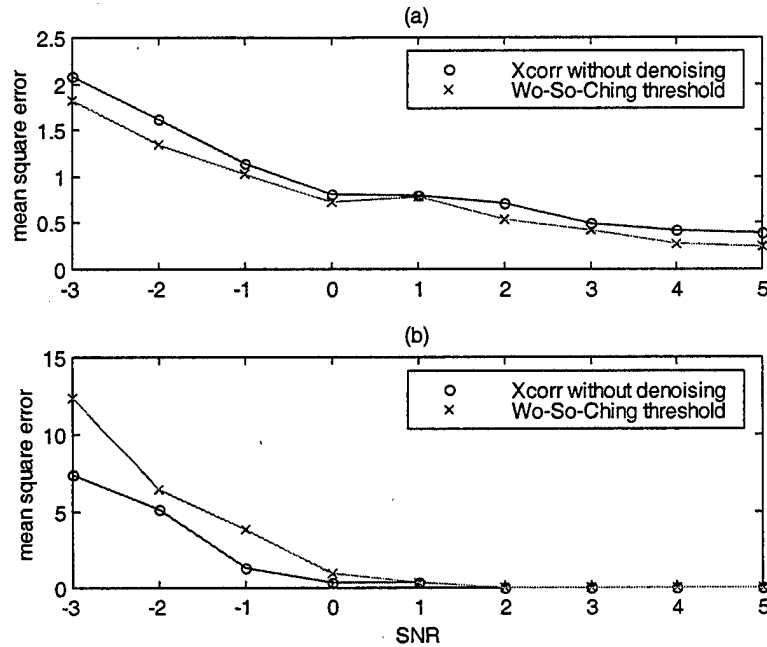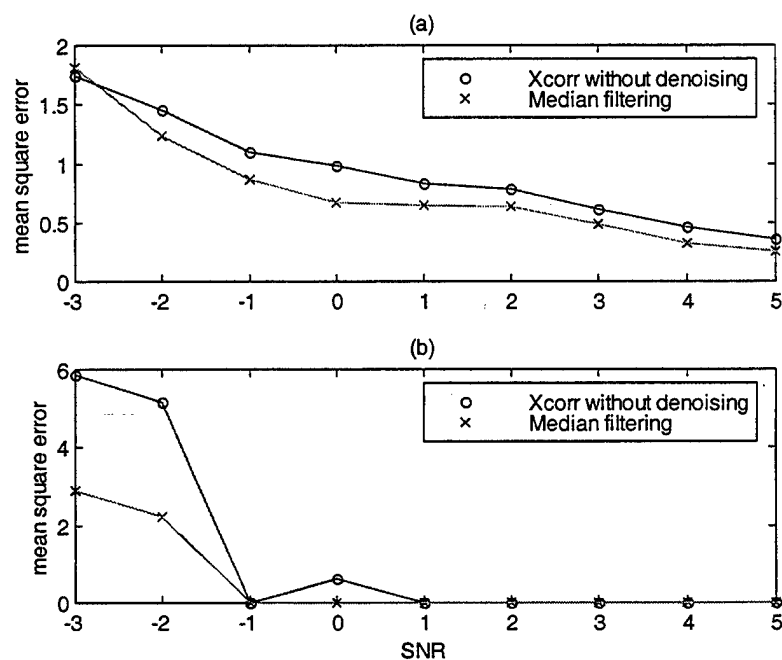

Figure 7.10: (a) MSE versus SNR with and without denoising using the median filtering method for generic signal A. (b) MSE for generic signal B.
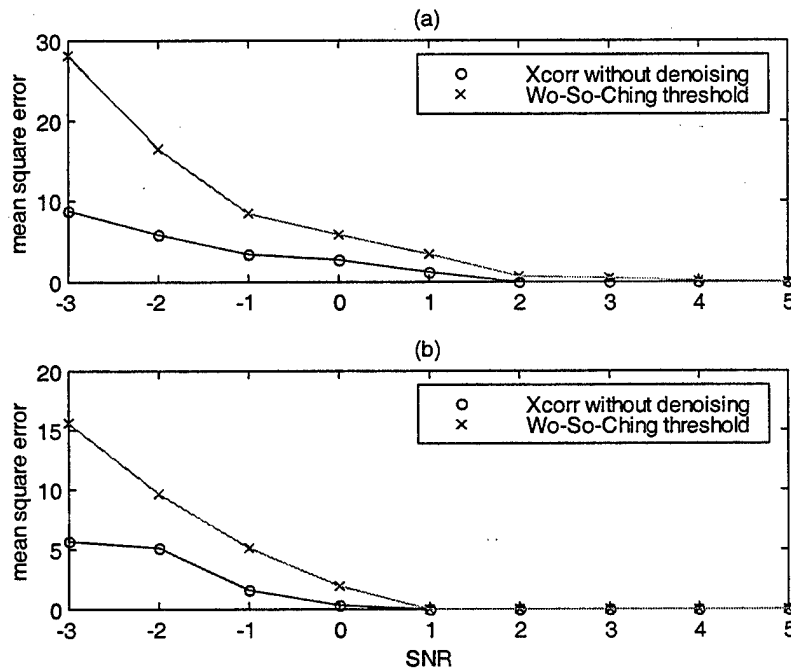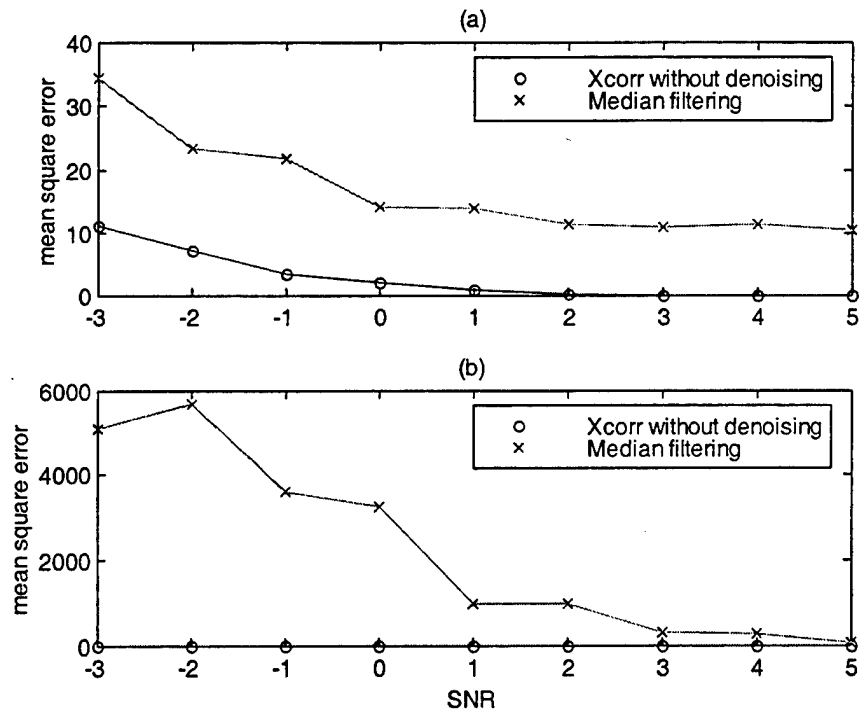
Figure 7.11: (a) MSE versus SNR with and without denoising using the median filtering method for generic signal C. (b) MSE for generic signal D.

SNR	Xcorr (A)	Median filtering (A)	Xcorr (B)	Median filtering (B)
5	0.375	0.26	0	0
4	0.53	0.325	0	0
3	0.52	0.49	0	0
2	0.625	0.635	0	0
1	0.96	0.645	0.32	0
0	1.07	0.675	0	0.005
-1	1.345	0.865	2.89	0.005
-2	1.36	1.235	4.505	2.24
-3	1.895	1.81	8.38	2.895
-4	2.425	2.125	12.025	8.855
-5	2.57	2.855	38.24	19.965
-6	87.41	28.35	268.01	200.06

Table 7.3: (a) MSE versus SNR for with and without denoising using the median filtering method for generic signals A and B.

SNR	Xcorr (C)	Median filtering (C)	Xcorr (D)	Median filtering (D)
5	0	10.52	0	67.555
4	0	11.32	0	264.57
3	0.08	10.84	0	301.49
2	0.16	11.32	0	963.45
1	1.04	13.88	0	973.63
0	1.6	14.11	0	3251.7
-1	4.08	21.825	1.645	3593.1
-2	4.08	23.425	0.96	5698.7
-3	9.76	34.415	8.76	5088.8
-4	14.8	178.33	15.385	9083.3
-5	24.96	385.47	22.97	9397.2
-6	118.8	1217.5	57.635	9916

Table 7.3: (b) MSE versus SNR for with and without denoising using the median filtering method for generic signals C and D.

b.   Simulation Results for GSM Signals

As shown in Figure 7.12, there is about 42% improvement in the total mean square error using the median filtering method for the GSM signal. If we compare the Figure 7.6, 7.9 and 7.12 we can conclude that hyperbolic shrinkage method gives the best result for the GSM signal.

5.   Modified Approximate Maximum-Likelihood Delay Estimation

a.   Simulation Results for the Generic Test Signals

As seen in Figure 7.13 and 7.14, there is a significant improvement in the mean square error for all generic signals when using the modified AML estimation method. The results are summarized in Table 7.4. If we compare the Table 7.1, 7.2, 7.3 and 7.4, we can conclude that the modified AML estimation method gives the best result.
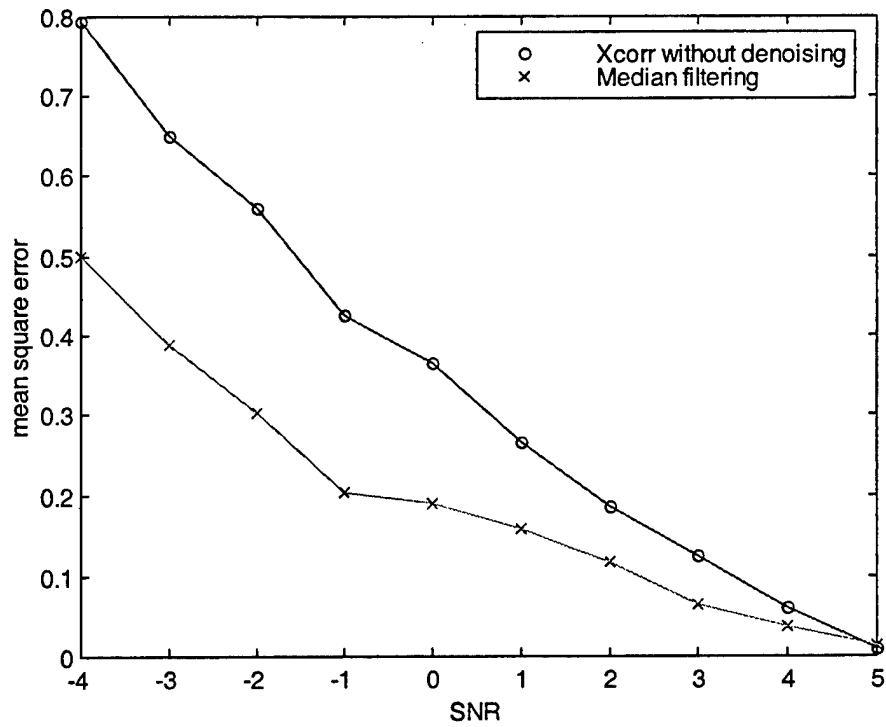
Figure 7.12: MSE versus SNR with and without denoising using the median filtering method for the GSM signal.
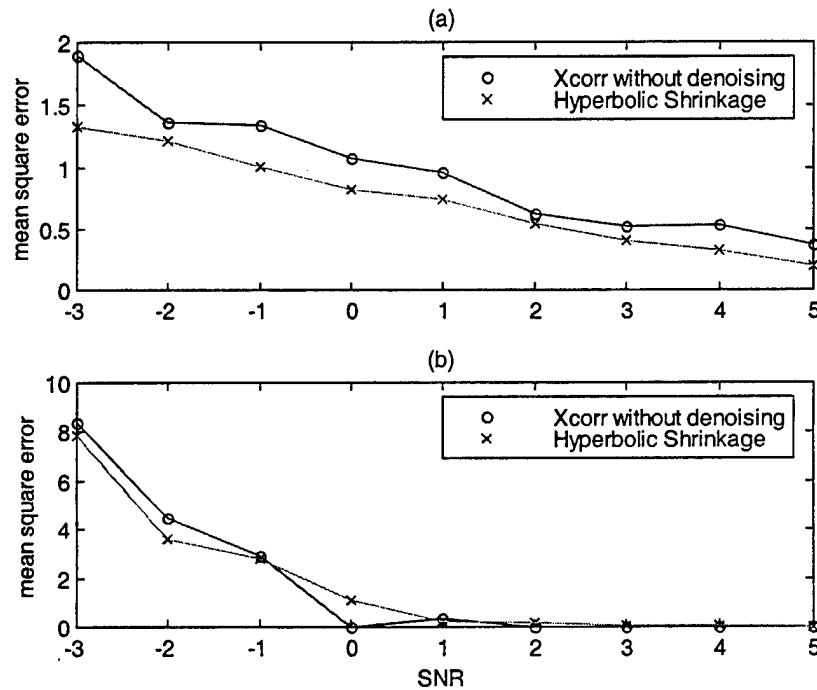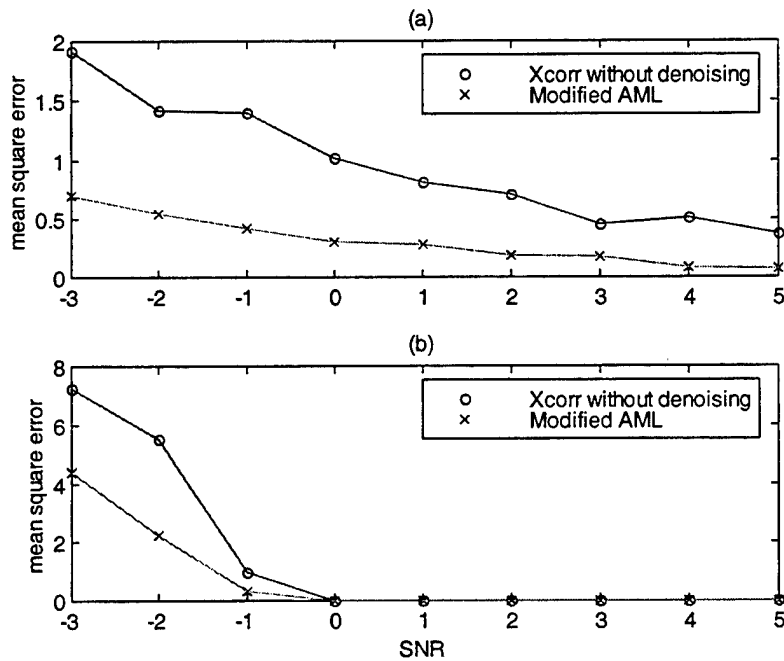

Figure 7.13: (a) MSE versus SNR with and without denoising using the modified AML estimation method for generic signal A. (b) MSE for generic signal B.
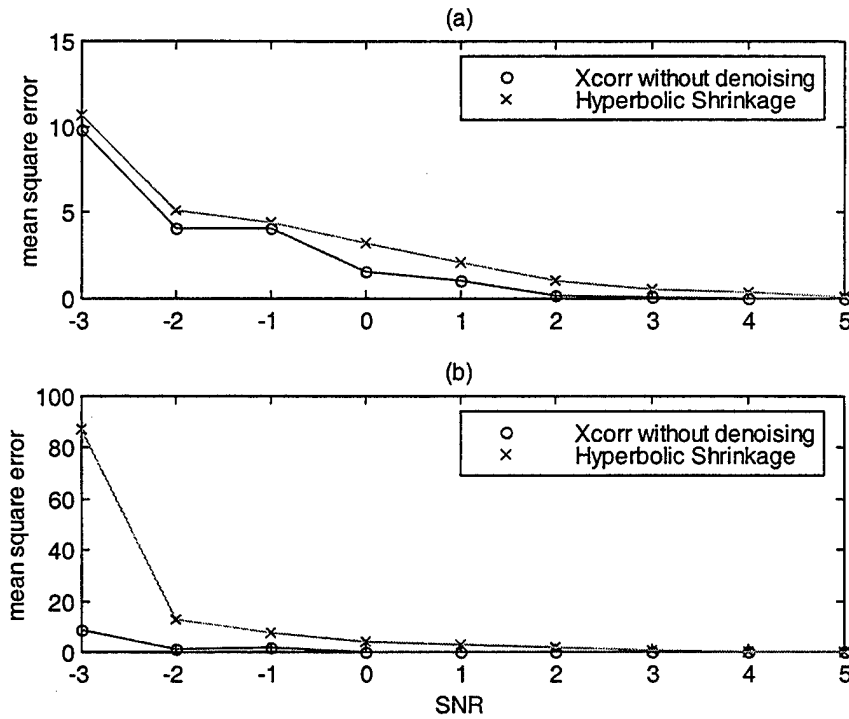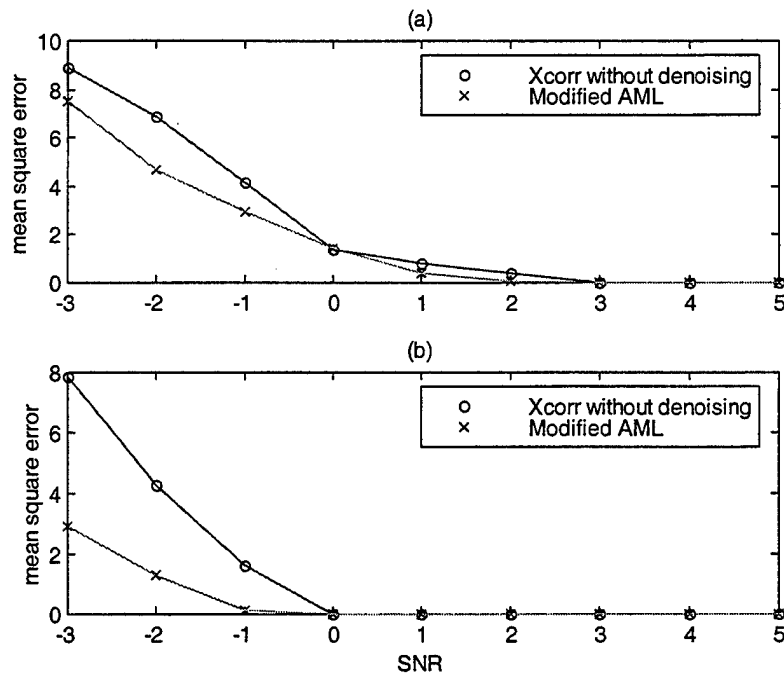
Figure 7.14: (a) MSE versus SNR with and without denoising using the modified AML estimation method for generic signal C. (b) MSE for generic signal D.

SNR	Xcorr (A)	Modified AML (A)	Xcorr (B)	Modified AML (B)
5	0.375	0.07	0	0
4	0.53	0.085	0	0
3	0.52	0.17	0	0
2	0.625	0.185	0	0
1	0.96	0.275	0.32	0
0	1.07	0.305	0	0
-1	1.345	0.42	2.89	0.325
-2	1.36	0.54	4.505	2.255
-3	1.895	0.695	8.38	4.43
-4	2.425	0.815	12.025	7.995
-5	2.57	1.26	38.24	13.085
-6	87.41	1.635	268.01	61.795

Table 7.4: (a) MSE versus SNR for with and without denoising the modified AML estimation method for generic signals A and B.

SNR	Xcorr (C)	Modified AML (C)	Xcorr (D)	Modified AML (D)
5	0	0	0	0
4	0	0	0	0
3	0.08	0	0	0
2	0.16	0.08	0	0
1	1.04	0.4	0	0
0	1.6	1.44	0	0
-1	4.08	2.96	1.645	0.125
-2	4.08	4.685	0.96	1.28
-3	9.76	7.525	8.76	2.925
-4	14.8	10.45	15.385	10.75
-5	24.96	20.395	22.97	13.35
-6	118.8	2670.8	57.635	208.15

Table 7.4: (b) MSE versus SNR for with and without denoising using the modified AML estimation method for generic signals C and D.

b.   Simulation Results for GSM Signals

As seen in Figure 7.15, there is about 79% improvement in the total mean square error using the modified AML estimation method relative to the undenoised version. Comparing Figure 7.6, 7.9, 7.12 and 7.15 we can conclude that the modified AML estimation method gives the best result for the GSM signal.

6.   Wavelet Denoising Based on the Fourth Order Moment

a.   Simulation Results for the Generic Signals

As seen in Figure 7.16 and 7.17, there is a significant improvement in the mean square error using *wavelet denoising based on the fourth order moment method* for all generic signals. The results are summarized in Table 7.5. If we compare this method with the other methods, we can conclude that wavelet denoising based on the fourth order moment is the second best method.

70
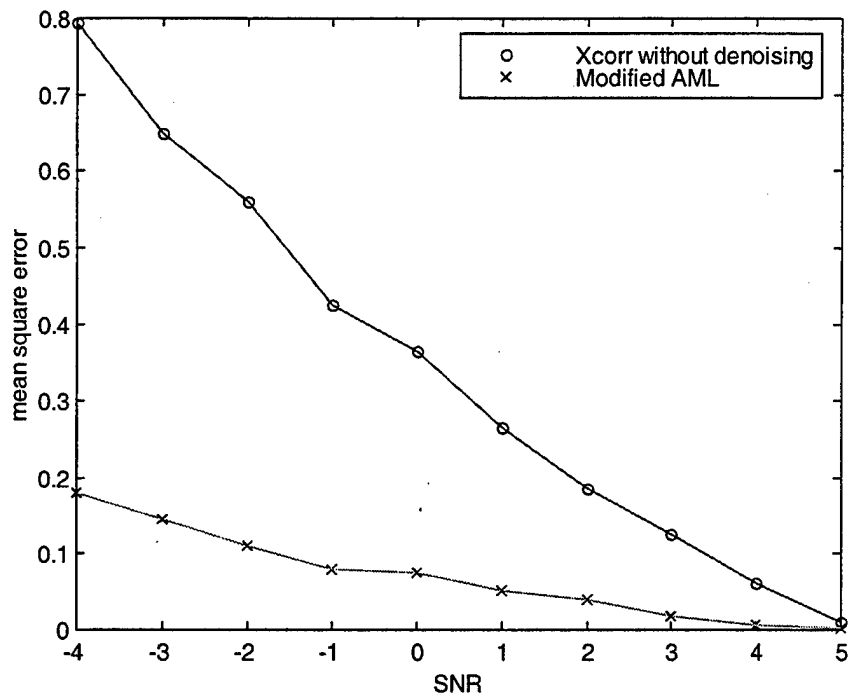
Figure 7.15: MSE versus SNR with and without denoising using the modified AML estimation method for the GSM signal.

Figure 7.16: (a) MSE versus SNR with and without denoising using the *wavelet denoising based on the fourth order moment method* method for generic signal A. (b) MSE for generic signal B.

Figure 7.17: (a) MSE versus SNR with and without denoising using the *wavelet denoising based on the fourth order moment method* method for generic signal C. (b) MSE for generic signal D.

SNR	Xcorr (A)	Fourth order (A)	Xcorr (B)	Fourth order (B)
5	0.375	0.08	0	0
4	0.53	0.11	0	0
3	0.52	0.16	0	0
2	0.625	0.215	0	0
1	0.96	0.35	0.32	0
0	1.07	0.47	0	0.32
-1	1.345	0.58	2.89	0.64
-2	1.36	0.82	4.505	1.6
-3	1.895	0.92	8.38	4.73
-4	2.425	1.455	12.025	5.775
-5	2.57	1.625	38.24	10.58
-6	87.41	2.54	268.01	21.2

Table 7.5: (a) MSE versus SNR for with and without denoising *wavelet denoising based on the fourth order moment* method for generic signals A and B.

72

SNR	Xcorr (C)	Fourth order (C)	Xcorr (D)	Fourth order (D)
5	0	0	0	0
4	0	0	0	0
3	0.08	0	0	0
2	0.16	0.24	0	0
1	1.04	0.72	0	0
0	1.6	1.12	0	0
-1	4.08	3.84	1.645	0.125
-2	4.08	5.52	0.96	1.92
-3	9.76	8.49	8.76	4.845
-4	14.8	12.375	15.385	12.19
-5	24.96	20.86	22.97	16.595
-6	118.8	26.99	57.635	24.85

Table 7.5: (b) MSE versus SNR for with and without denoising using *wavelet denoising based on the fourth order moment* method for generic signals C and D.

b. Simulation Results for GSM Signals

As seen in Figure 7.18, there is about 63% improvement in the mean square error using *wavelet denoising based on the fourth order moment* method. If we compare Figure 7.6, 7.9, 7.12, 7.15 and 7.18 we can conclude that *wavelet denoising based on the fourth order moment* ranks second in performance for the GSM signal.

7. Time Varying Technique

As we explained in Chapter VI, we can address signals that are non-stationary (i.e., the signal does not stay in a given frequency band for the length of the segment or has modulation characteristics). In principle, in any given subband we try to find several weights. This allows us to keep the information for the frequency band when the signal is strong. We tried this method on the GSM signal and compared it with the modified AML estimation method. Figure 7.19 allows a comparison of the modified AML method with the time varying version in which the segments are partitioned into two parts for every

scale. The improvement in the total mean square error using the time varying modified AML method is about 81%. There is no drastic improvement relative to the modified AML, but we believe that if the signal has a time varying property, the new method based on the time varying technique will give better results.

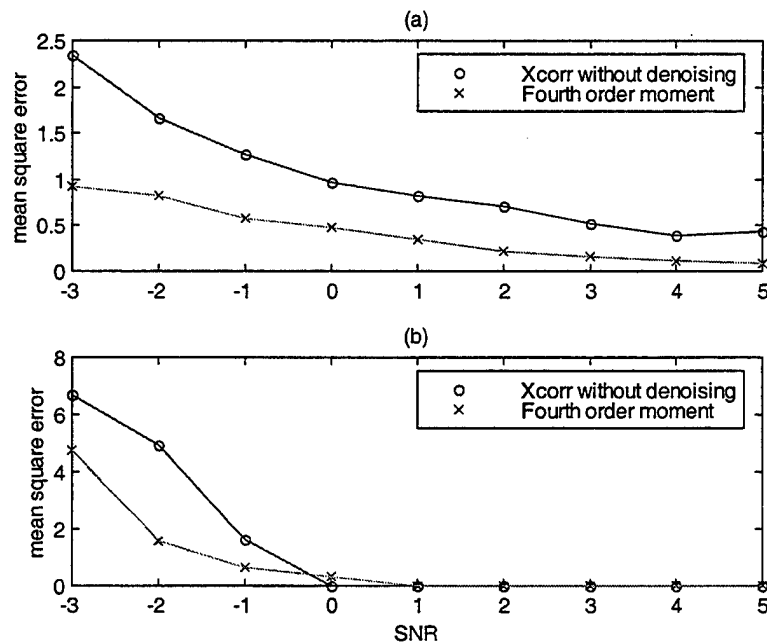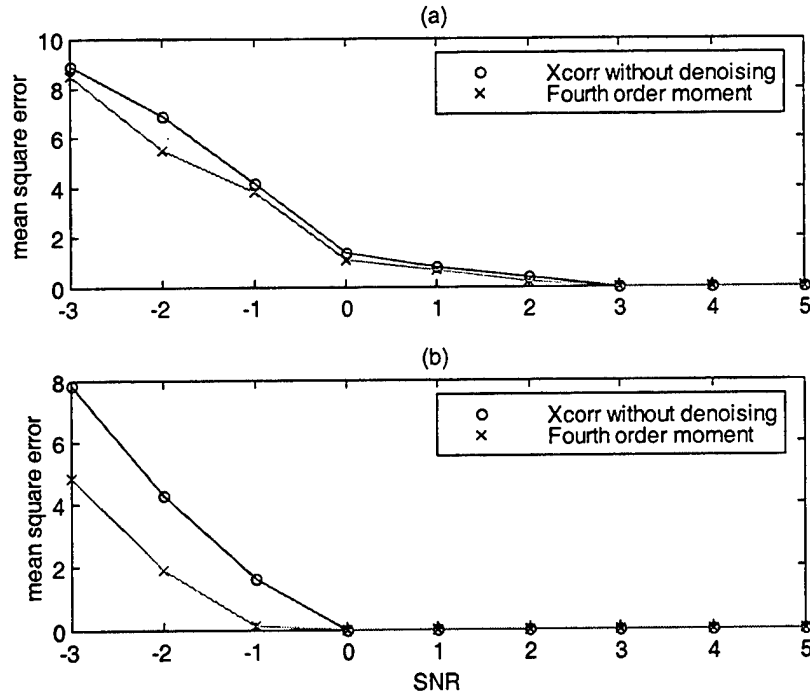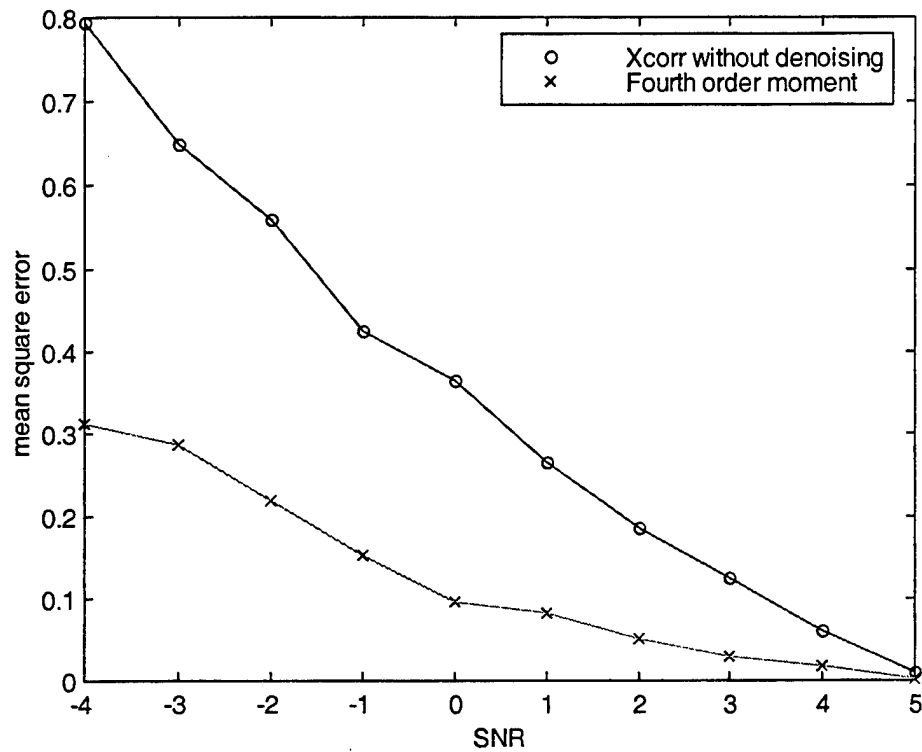Figure 7.18: MSE versus SNR with and without denoising using the *wavelet denoising based on the fourth order moment* method for the GSM signal.
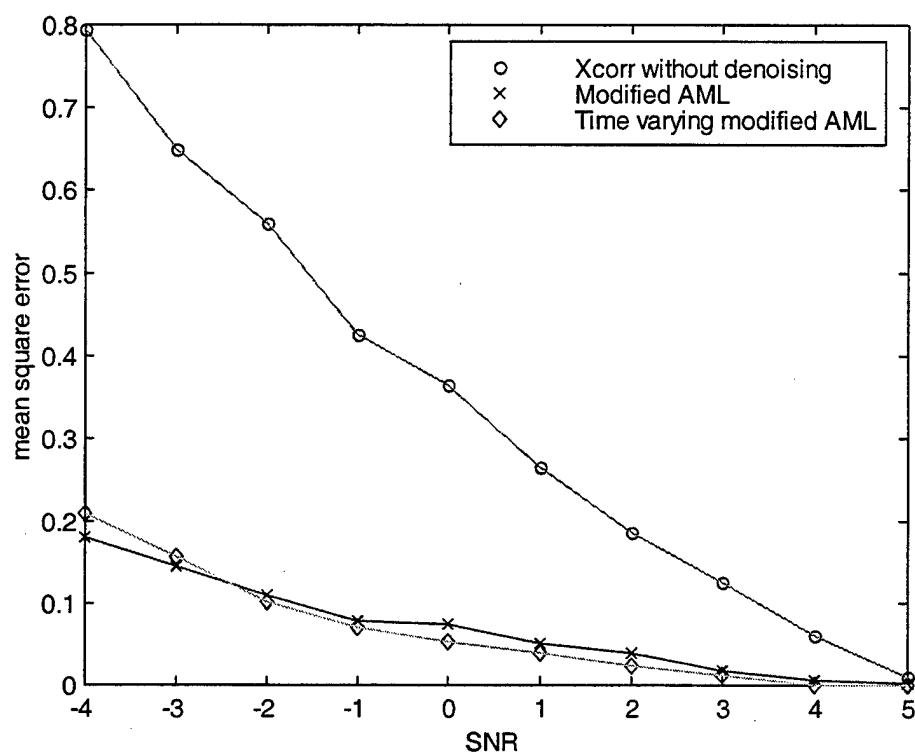
Figure 7.19: MSE versus SNR with and without denoising using the *time varying modified AML* and *modified AML technique* on the GSM signal.

VIII. CONCLUSION AND FUTURE WORK

A. CONCLUSION

The accuracy of algorithms used, to localize wireless communication units, was studied. Chapter I reviewed why there is a need to locate cellular transmitters and the existing localization systems. The time difference of arrival (TDOA) method was used to locate emitters. Estimation of the TDOA was based on the cross correlation function. To increase the accuracy of the TDOA estimation, the noise in the received data was reduced. The wavelet transform was employed to minimize the noise, and several denoising methods were examined.

The MSE's for all methods for the GSM signal were plotted in Figure 8.1. The *modified AML delay estimation* method provided the best result, *wavelet denoising using the fourth order moment* method ranked second. All methods provided better results than the correlation of the raw time domain signals. The time varying technique outperformed the modified AML method for SNR values greater or equal to –2 dB. We believe that if the signal has time varying properties, the time varying technique will be the superior method.

The probability of no-error for the GSM signal was plotted in Figure 8.2 and 8.3. The Wo-So-Ching, the hyperbolic shrinkage and the median filtering techniques, see Figure 8.2, gave worse results than the correlation of the raw time domain signals. In these techniques, a threshold value was calculated and then according to the threshold, each coefficient was modified. These three techniques can modify the coefficients, which

77

represent the signal in the subband. Hence we reduced the mean square error by using these three techniques but we also decreased the probability of no-error.

The modified AML, the fourth order moment and the time varying AML techniques, see Figure 8.3, gave better results than the correlation of the raw time domain signals. In these techniques, we evaluated the whole subband, The subband or portion of it, which represented the signal was kept. The subband, which represented noise only was eliminated. Since there was a small chance to modify signal coefficients representing the signal, these three techniques improved the probability of no-error. The time varying AML method gave the highest probability of no-error.
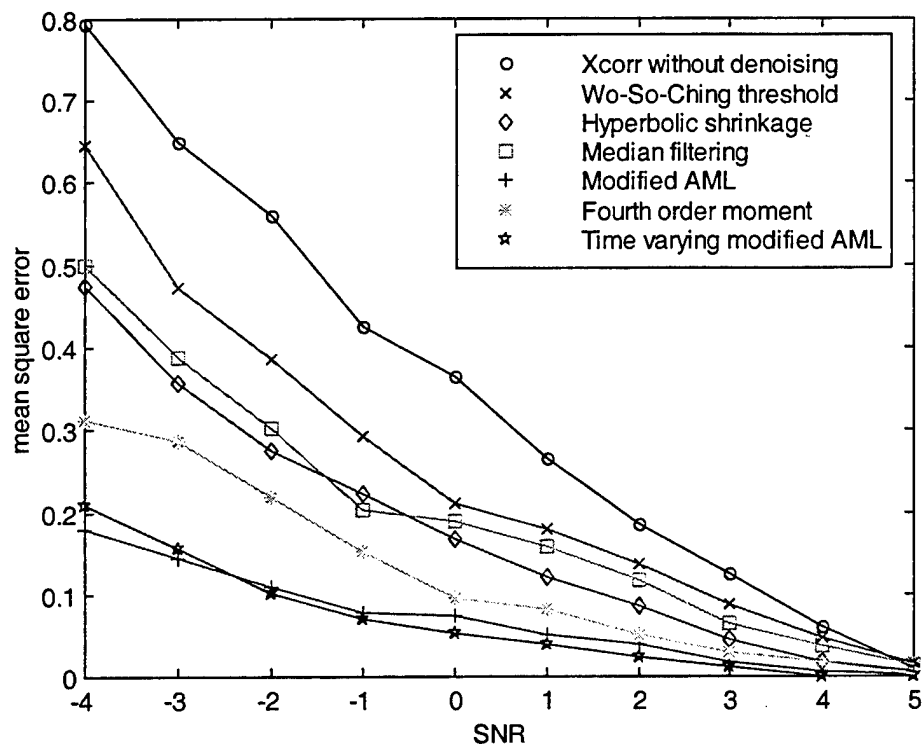
Figure 8.1: Plot of all denoising methods for the GSM signal.
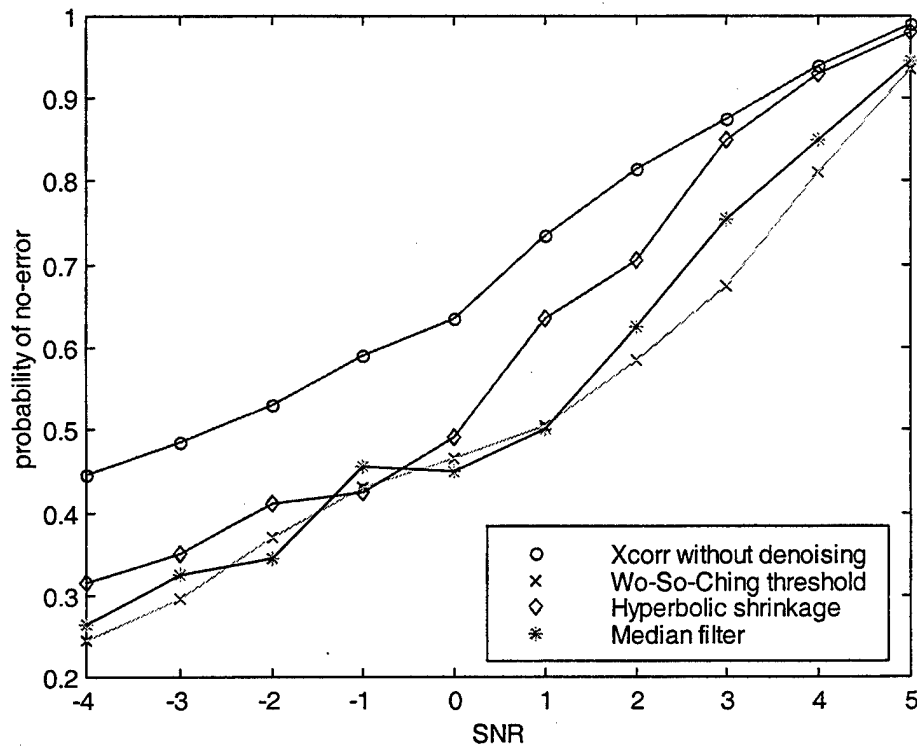
Figure 8.2: Plot of the probability of no-error for the Wo-So-Ching threshold, Hyperbolic Shrinkage and Median filtering techniques.

B. FUTURE WORK

1. We worked exclusively with the *base band* GSM signal. A follow on study should address GSM signals at the RF or IF level, to evaluate if there is potential improvement of the MSE.

2. In our experiments, only white Gaussian noise was simulated. Fading signals should be included in a follow on study.
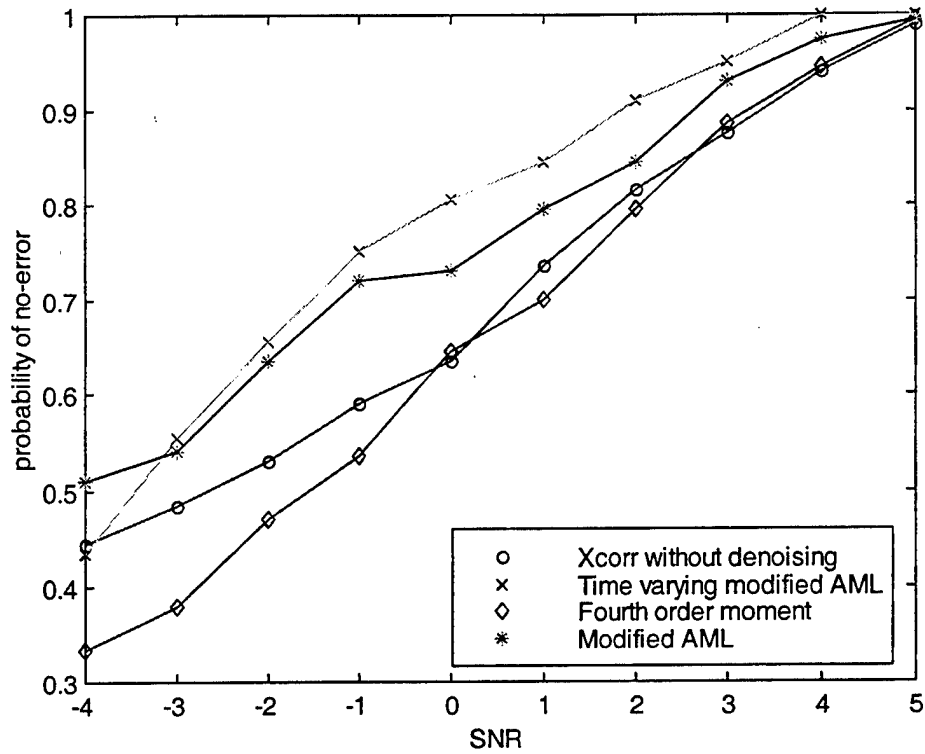
Figure 8.3: Plot of the probability of no-error for the Time varying AML, Fourth order moment and Median filtering techniques.

APPENDIX

MATLAB CODES

A.   SIGNAL GENERATION

1.   Generic Signal

```
%***************************************************************************
%***************************************************************************
% u_mod: This function creates generic signals
%           f=1 for Test signal A
%           f=4 for Test signal B
%           f=8 for Test signal C
%           f=12 for Test signal D
%
% SYNTAX: y=u_mod(data)
%
% INPUT: data = Digital data stream
%
% OUTPUT: y = PSK modulated data
%
% SUB_FUNC: None
% Written by Unal Aktas
%***************************************************************************
%***************************************************************************


function y=u_mod(data);

n=0:31;  % NUMBER OF SAMPLES
one=sin(2*pi*n*f*/32);

% f=1 for Test signal A
% f=4 for Test signal B
% f=8 for Test signal C
% f=12 for Test signal D

zero=-1.*one;
mod_data=[];
for i=1:length(data)
   if data(i)==1
      mod_data=[mod_data one];
   else
      mod_data=[mod_data zero];
   end

end
plot(mod_data);title('modulated signal')
y=mod_data;
```

2.   GSM Signal

```
%***********************************************************************
%***********************************************************************
% GSM_SET:  This script initializes the values.
%
% SYNTAX:   gsm_set
%
% INPUT:    None
% OUTPUT:   Configuration variables created in memory, these are:
%           Tb(= 3.692e-6)
%           BT(= 0.3)
%           OSR(= 4)
%           SEED(= 931316785)
%           INIT_L(= 260)
%
% SUB_FUN:  None
% Written by Jan H. Mikkelsen / Arne Norre Ekstrom
%***********************************************************************
%***********************************************************************

Tb = 3.692e-6;
BT = 0.3;
OSR = 4;

% INITIALIZE THE RANDOM NUMBER GENERATOR.
% BY USING THE SAME SEED VALUE IN EVERY SIMULATION, WE GET THE SAME
% SIMULATION DATA, AND THUS SIMULATION RESULTS MAY BE REPRODUCED.
%
SEED = 931316785;
rand('seed',SEED);

% THE NUMBER OF BITS GENERATED BY THE DATA GENERATOR. (data_gen)
%
INIT_L = 114;

% SETUP THE TRAINING SEQUENCE USED FOR BUILDING BURSTS
%
TRAINING = [0 0 1 0 0 1 0 1 1 1 0 0 0 0 1 0 0 0 1 0 0 1 0 1 1 1];

% CONSTRUCT THE MSK MAPPED TRAINING SEQUENCE USING TRAINING.
%
T_SEQ = T_SEQ_gen(TRAINING);
```

```
%*******************************************************************
%*******************************************************************
% burst_g:   This function generates a bit sequence representing
%            a general GSM information burst. Included are tail
%            and ctrl bits, data bits and a training sequence.
%
%            The GSM burst contains a total of 148 bits accounted
%            for in the following burststructure (GSM 05.05)
%
%            [ TAIL | DATA | CTRL | TRAINING | CTRL | DATA | TAIL ]
%            [  3    |  57  |  1   |    26    |  1   |  57  |  3   ]
%
%            [TAIL]    = [000]
%            [CTRL]    = [0] or [1] here [1]
%            [TRAINING] is passed to the function
%
% SYNTAX:    tx_burst = burst_g(tx_data, TRAINING)
%
% INPUT:     tx_data:  The burst data.
%            TRAINING: The Training sequence which is to be used.
%
% OUTPUT:    tx_burst: A complete 148 bits long GSM normal burst binary
%                      sequence
%
% GLOBAL:
%
% SUB_FUNC: None
% Written by Jan H. Mikkelsen / Arne Norre Ekstrom
%*******************************************************************
%*******************************************************************

function tx_burst = burst_g(tx_data, TRAINING)

TAIL = [0 0 0];
CTRL = [1];

% COMBINE THE BURST BIT SEQUENCE
%
tx_burst = [TAIL tx_data(1:57) CTRL TRAINING CTRL tx_data(58:114)
TAIL];
```

```
%*************************************************************
%*************************************************************
%
% data_gen:    This function generates a block of random data bits.
%
% SYNTAX:      [tx_data] = data_gen(INIT_L)
%
% INPUT:       INIT_L:  The length of the generated data vector.
%
% OUTPUT:      tx_data: An element vector contaning the random data
%                       sequence of length INIT_L. INIT_L is a variable
%                       set by gsm_set.
%
% SUB_FUNC:    None
% Written by Jan H. Mikkelsen / Arne Norre Ekstrom
%*************************************************************
%*************************************************************

function [tx_data] = data_gen(INIT_L);

tx_data = round(rand(1,INIT_L));
```

```
%****************************************************************
%****************************************************************
% GSM_MOD:   This MatLab code generates a GSM normal burst by
%            combining tail, ctrl, and training sequence bits with
%            two bloks of random data bits.
%            The data bits are convolutional encoded according
%            to the GSM recommendations
%            The burst sequence is differential encoded and then
%            subsequently GMSK modulated to provide oversampled
%            I and Q baseband representations.
%
% SYNTAX:    [ tx_burst , I , Q ] = gsm_mod(Tb,OSR,BT,tx_data,TRAINING)
%
% INPUT:     Tb: Bit time, set by gsm_set.m
%           OSR: Oversampling ratio (fs/rb), set by gsm_set.m
%            BT: Bandwidth Bittime product, set by gsm_set.m
%       tx_data: The contents of the datafields in the burst to be
%                transmitted. Datafield one comes first.
%      TRAINING: The Training sequence which is to be inserted in the
%                burst.
%
% OUTPUT:
%      tx_burst: The entire transmitted burst before differential
%                precoding.
%             I: Inphase part of modulated burst.
%             Q: Quadrature part of modulated burst.
% Written by Jan H. Mikkelsen / Arne Norre Ekstrom
%****************************************************************
%****************************************************************

function [ tx_burst , I , Q ] = gsm_mod(Tb,OSR,BT,tx_data,TRAINING)

tx_burst = burst_g(tx_data,TRAINING);

% DIFFERENTIAL ENCODING.
%
burst = diff_enc(tx_burst);

% GMSK MODULATION OF BURST SEQUENCE
%
[I,Q] = gmsk_mod(burst,Tb,OSR,BT);
```

```
%*********************************************************************
%*********************************************************************
% diff_enc:   This function accepts a GSM burst bit sequence and
%             performs a differential encoding of the sequence. The
%             encoding is according to the GSM 05.05 recommendations
%
%             Input: D(i)
%             Output: A(i)
%
%             D'(I) = D(i) (+) D(i-1)   ; (+) = XOR
%                D(i), D'(i) = {0,1}
%             A(i) = 1 - 2*D'(i)
%                A(i) = {-1,1}
%
% SYNTAX:     [diff_enc_data] = diff_enc(burst)
%
% INPUT:      burst   A binary, (0,1), bit sequence
%
% OUTPUT:     diff_enc_data   A differential encoded, (+1,-1), version
%              of the input burst sequence
%
% SUB_FUNC:   None
% Written by Jan H. Mikkelsen / Arne Norre Ekstrom
%*********************************************************************
%*********************************************************************
function DIFF_ENC_DATA = diff_enc(BURST)

L = length(BURST);

% INTERMEDIATE VECTORS FOR DATA PROCESSING
%
d_hat = zeros(1,L);
alpha = zeros(1,L);

% DIFFERENTIAL ENCODING ACCORDING TO GSM 05.05
% AN INFINITE SEQUENCE OF 1'ENS ARE ASSUMED TO
% PRECEED THE ACTUAL BURST
%
data = [1 BURST];
for n = 1+1:(L+1),
  d_hat(n-1) = xor( data(n),data(n-1) );
end
alpha = 1 - 2.*d_hat;

% PREPARING DATA FOR OUTPUT
%
DIFF_ENC_DATA = alpha;
```

```
%*******************************************************************
%*******************************************************************
%
% gmsk_mod:    This function accepts a GSM burst bit sequence and
%              performs a GMSK modulation of the sequence. The
%              modulation is according to the GSM 05.05 recommendations
%
% SYNTAX:      [i,q] = gmsk_mod(burst,Tb,osr,BT)
%
% INPUT:       burst   A differential encoded bit sequence (-1,+1)
%              Tb      Bit duration (GSM: Tb = 3.692e-6 Sec.)
%              osr     Simulation oversample ratio. osr determines the
%                      number of simulation steps per information bit
%              BT      The bandwidth/bit duration product (GSM: BT =
% 0.3)
%
% OUTPUT:      i,q     In-phase (i) and quadrature-phase (q) baseband
%                      representation of the GMSK modulated input burst
%                      sequence
%
% SUB_FUNC:    ph_g.m  This sub-function is required in generating the
%                      frequency and phase pulse functions.
% Written by Jan H. Mikkelsen / Arne Norre Ekstrom
%*******************************************************************
%*******************************************************************

function [I,Q] = gmsk_mod(BURST,Tb,OSR,BT)
[g,q] = ph_g(Tb,OSR,BT);

% PREPARE VECTOR FOR DATA PROCESSING
%
bits = length(BURST);
f_res = zeros(1,(bits+2)*OSR);

% GENERATE RESULTING FREQUENCY PULSE SEQUENCE
%
for n = 1:bits,
   f_res((n-1)*OSR+1:(n+2)*OSR) = f_res((n-1)*OSR+1:(n+2)*OSR) +
BURST(n).*g;
end

% CALCULATE RESULTING PHASE FUNCTION
%
theta = pi*cumsum(f_res);

% PREPARE DATA FOR OUTPUT
%
I = cos(theta);
Q = sin(theta);
```

```
%************************************************************************
%************************************************************************
%
% PH_G:      This function calculates the frequency and phase functions
%            required for the GMSK modulation. The functions are
%            generated according to the GSM 05.05 recommendations
%
% SYNTAX:    [g_fun, q_fun] = ph_g(Tb,osr,BT)
%
% INPUT:     Tb      Bit duration (GSM: Tb = 3.692e-6 Sec.)
%            osr     Simulation oversample ratio. osr determines the
%                    number of simulation steps per information bit
%            BT      The bandwidth/bit duration product (GSM: BT = 0.3)
%
% OUTPUT:    g_fun, q_fun   Vectors contaning frequency and phase
%                           function outputs when evaluated at osr*tb
%
% SUB_FUNC: None
% Written by Jan H. Mikkelsen / Arne Norre Ekstrom
%************************************************************************
%************************************************************************

function [G_FUN, Q_FUN] = ph_g(Tb,OSR,BT)
Ts = Tb/OSR;

% PREPARING VECTORS FOR DATA PROCESSING
%
PTV = -2*Tb:Ts:2*Tb;
RTV = -Tb/2:Ts:Tb/2-Ts;

% GENERATE GAUSSIAN SHAPED PULSE
%
sigma = sqrt(log(2))/(2*pi*BT);
gauss = (1/(sqrt(2*pi)*sigma*Tb))*exp(-PTV.^2/(2*sigma^2*Tb^2));

% GENERATE RECTANGULAR PULSE
%
rect = 1/(2*Tb)*ones(size(RTV));

% CALCULATE RESULTING FREQUENCY PULSE
%
G_TEMP = conv(gauss,rect);

% TRUNCATING THE FUNCTION TO 3xTb
%
G = G_TEMP(OSR+1:4*OSR);

% TRUNCATION IMPLIES THAT INTEGRATING THE FREQUENCY PULSE
% FUNCTION WILL NOT EQUAL 0.5, HENCE THE RE-NORMALIZATION
%
G_FUN = (G-G(1))./(2*sum(G-G(1)));

% CALCULATE RESULTING PHASE PULSE
%
Q_FUN = cumsum(G_FUN);
```

B. DENOISING TECHNIQUES

1. Modified AML

```
%*****************************************************************
%*****************************************************************
% aml1: Approximate maximum-likelihood delay estimation
%        via orthogonal wavelet transform. In this function,
%        we assumed that noise is Gaussian noise and it has a
%        flat freq response. We modify each detail function
%        by multipliying modified AML coefficient based on
%        the signal and noise powers
%
% SYNTAX: y=aml1(xn,yn,delay)
%
% INPUT: xn = Received signal from first receiver
%        yn = Received signal from second receiver
%     delay = True TDOA between two received signals
%
% OUTPUT:
%        y = Error between true TDOA and estimated TDOA
%
% SUB_FUNC: None
% Written by Unal Aktas
%*****************************************************************
%*****************************************************************

function y=aml1(xn,yn,delay);

xyn=xcorr(xn,yn,'biased');
[sigmas b]=max(xyn);
rx=xcorr(xn,'biased');
maxx=rx(length(xn));
ry=xcorr(yn,'biased');
maxy=ry(length(yn));
sigman1=maxx-sigmas;
sigman2=maxy-sigmas;

nx=floor(log2(length(xn)));
ny=floor(log2(length(yn)));

[cx lx]=wavedec(xn,nx,'db4');
[cy ly]=wavedec(yn,ny,'db4');

dxc=[];
for i=1:nx
   d=detcoef(cx,lx,i);
   dl=length(d);
   sigmad=(1/dl)*sum(d.^2);
   sigmasd=sigmad-sigman1;
   if sigmasd<=0
      wd=0;
   else
      wd=sigmasd/(sigman1*sigman2+sigmasd*(sigman1+sigman2));
   end
```

```
      dc=wd*d;
      dxc=[dc dxc];
end

a=appcoef(cx,lx,'db4',nx);
al=length(a);
sigmaa=(1/al)*sum(a.^2);
sigmasa=sigmaa-sigman1;
if sigmasa<=0
    wa=0;
else
    wa=sigmasa/(sigman1*sigman2+sigmasa*(sigman1+sigman2));
end
ac=wa*a;
dxc=[ac dxc];

xd=waverec(dxc,lx,'db4');

dyc=[];
for i=1:ny
    dy=detcoef(cy,ly,i);
    dyl=length(dy);
    sigmady=(1/dyl)*sum(dy.^2);
    sigmasdy=sigmady-sigman1;
    if sigmasdy<=0;
        wdy=0;
    else
        wdy=sigmasdy/(sigman1*sigman2+sigmasdy*(sigman1+sigman2));
    end
    dcy=wdy*dy;
    dyc=[dcy dyc];
end

ay=appcoef(cy,ly,'db4',ny);
ayl=length(ay);
sigmaay=(1/ayl)*sum(ay.^2);
sigmasay=sigmaay-sigman1;
if sigmasay<=0
    way=0;
else
    way=sigmasay/(sigman1*sigman2+sigmasay*(sigman1+sigman2));
end
acy=way*ay;
dyc=[acy dyc];

yd=waverec(dyc,ly,'db4');

rxyd=xcorr(xd,yd);

[a5 b5]=max(rxyd);

er5=delay-(b5-1024);
y=er5;
```

```
%********************************************************************
%********************************************************************
%
% tez5: This is a test program for modified AML technique. In this
%       program, we used GSM signal.
%
% SYNTAX: tez5
%
% INPUT: None
%
% OUTPUT: Mean sqaure error versus SNR
%
% SUB_FUN: aml1.m
% Written by Unal Aktas
%********************************************************************
%********************************************************************
clear all
gsm_set;

data=data_gen(INIT_L); % this creates a binary data
[tx_burst,I,Q]=gsm_mod(Tb,OSR,BT,data,TRAINING);

s=I+j*Q;
sl=length(s);
pow=(1/sl)*sum(abs(s).^2);

K=200     % NUMBER OF REALIZATIONS
rand('seed',40);
f=150*rand(1,K);
delay=floor(f);
delay(1:K/2)=-delay(1:K/2);   % DELAY IS BETWEEN -150 TO +150

n=[5 4 3 2 1 0 -1 -2 -3 -4 -5 -6];
SNR=10.^(n./10);

for k=1:length(SNR)

    oran=SNR(k)
    for i=1:K

        x=[zeros(1,200) s zeros(1,224)];
        y=[zeros(1,200+delay(i)) s zeros(1,224-delay(i))];

        randn('state',2*(i+j));
        noi1_real=sqrt(pow/(2*oran))*randn(1,1024);

        randn('state',3*(i+j));
        noi1_imag=sqrt(pow/(2*oran))*randn(1,1024);

        randn('state',4*(i+j));
        noi2_real=sqrt(pow/(2*oran))*randn(1,1024);

        randn('state',5*(i+j));
        noi2_imag=sqrt(pow/(2*oran))*randn(1,1024);

        noi1=noi1_real+j*noi1_imag;
        noi2=noi2_real+j*noi2_imag;
```

```
    xn=x+noi1; % x + noise
    yn=y+noi2; % y + noise

    % TDOA CALCULATION WITHOUT DENOISING

    xy=xcorr(xn,yn);
    [a1 b1]=max(real(xy));

    er1(i)=delay(i)-(b1-1024);

    %modified AML

    e1=aml1(real(xn),real(yn),delay(i));
    e2=aml1(imag(xn),imag(yn),delay(i));

    er8(i)=(e1+e2)/2;

  end

  error8a(k)=(1/length(er8))*sum(er8.^2);
  error1a(k)=(1/length(er1))*sum(er1.^2);

  H8a(k,:)=er8;
  H1a(k,:)=er1;

end

figure(6)
k=[5 4 3 2 1 0 -1 -2 -3 -4];
plot(k,error1a(1:10),'o',k,error8a(1:10),'x',k,error1a(1:10),k,...
    error8a(1:10))
legend('xcorr without denoising','modified AML')

save error8a;
save error1a;

save H1a;
save H8a;
```

2.   Time Varying MAML

```
%*****************************************************************
%*****************************************************************
% aml2: We modified the MAML technique by dividing each
%         detail function into two segments. Different
%         coefficients for each segment are computed.
%
% SYNTAX: y=aml2(xn,yn,delay)
%
% INPUT: xn = Received signal from first receiver
%          yn = Received signal from second receiver
%       delay = True TDOA between xn and yn
% OUTPUT: y = Error between true TDOA and estimated TDOA
%
% SUB_FUNC: None
% Written by Unal Aktas
%*****************************************************************
%*****************************************************************


function y=aml2(xn,yn,delay);

xyn=xcorr(xn,yn,'biased');
[sigmas b]=max(xyn);
rx=xcorr(xn,'biased');
maxx=rx(length(xn));
ry=xcorr(yn,'biased');
maxy=ry(length(yn));
sigman1=maxx-sigmas;
sigman2=maxy-sigmas;

nx=floor(log2(length(xn)));
ny=floor(log2(length(yn)));

[cx lx]=wavedec(xn,nx,'db4');
[cy ly]=wavedec(yn,ny,'db4');

dxc=[];
for i=1:nx
    d=detcoef(cx,lx,i);
    dl=length(d);
    Ns1=128/2^(i-1);  % THE LENGTH OF THE SUBBLOCK
    if Ns1<=1
        Ns=dl;
    else
        Ns=Ns1;
    end

    D=ceil(dl/Ns);
    if dl<Ns*D
        dm=[d zeros(1,D*Ns-dl)];
    end
    for k=1:D
        p=(k-1)*Ns+1:k*Ns;
        sigmad=(1/Ns)*sum(dm(p).^2);
        sigmasd=sigmad-sigman1;
```

```
        if sigmasd<=0
            wd=0;
        else
            wd=sigmasd/(sigman1*sigman2+sigmasd*(sigman1+sigman2));
        end
        dc(p)=wd*dm(p);
    end
    dxc=[dc(1:dl) dxc];
end

a=appcoef(cx,lx,'db4',nx);
al=length(a);
sigmaa=(1/al)*sum(a.^2);
sigmasa=sigmaa-sigman1;
if sigmasa<=0
    wa=0;
else
    wa=sigmasa/(sigman1*sigman2+sigmasa*(sigman1+sigman2));
end
ac=wa*a;
dxc=[ac dxc];

xd=waverec(dxc,lx,'db4');

dyc=[];
for i=1:ny
    dy=detcoef(cy,ly,i);
    dyl=length(dy);
    Ns1=128/2^(i-1);
    if Ns1<=1
        Ns=dyl;
    else
        Ns=Ns1;
    end

    D=ceil(dyl/Ns);
    if dyl<Ns*D
        dmy=[dy zeros(1,D*Ns)];
    end
    for k=1:D
        p=(k-1)*Ns+1:k*Ns;
        sigmady=(1/Ns)*sum(dmy(p).^2);
        sigmasdy=sigmady-sigman1;
        if sigmasdy<=0;
            wdy=0;
        else
            wdy=sigmasdy/(sigman1*sigman2+sigmasdy*(sigman1+sigman2));
        end
        dcy(p)=wdy*dmy(p);
    end

    dyc=[dcy(1:dyl) dyc];
end

ay=appcoef(cy,ly,'db4',ny);
ayl=length(ay);
sigmaay=(1/ayl)*sum(ay.^2);
```

94

```
sigmasay=sigmaay-sigman1;
if sigmasay<=0
    way=0;
else
    way=sigmasay/(sigman1*sigman2+sigmasay*(sigman1+sigman2));
end
acy=way*ay;
dyc=[acy dyc];

yd=waverec(dyc,ly,'db4');

rxyd=xcorr(xd,yd);

[a5 b5]=max(rxyd);

er5=delay-(b5-1024);
y=er5;
```

```
%*****************************************************************
%*****************************************************************
% tez5t : This is a test program for time varying MAML technique.
%          In this program, GSM signal is used
%
% SYNTAX: tez5t
%
% INPUT: None
%
% OUTPUT: Mean square error versus SNR
%
% SUB_FUNC: aml2.m
% Written by Unal Aktas
%*****************************************************************
%*****************************************************************
clear all

gsm_set;

data=data_gen(INIT_L);
[tx_burst,I,Q]=gsm_mod(Tb,OSR,BT,data,TRAINING);

s=I+j*Q;
sl=length(s);
pow=(1/sl)*sum(abs(s).^2);

K=200      % NUMBER OF REALIZATIONS
rand('seed',40);
f=150*rand(1,K);
delay=floor(f);
delay(1:K/2)=-delay(1:K/2);   % DELAY IS BETWEEN -150 TO +150

n=[5 4 3 2 1 0 -1 -2 -3 -4 -5 -6];
SNR=10.^(n./10);

for k=1:length(SNR)

   oran=SNR(k)
   for i=1:K

      x=[zeros(1,200) s zeros(1,224)];
      y=[zeros(1,200+delay(i)) s zeros(1,224-delay(i))];

      randn('state',2*(i+j));
      noi1_real=sqrt(pow/(2*oran))*randn(1,1024);

      randn('state',3*(i+j));
      noi1_imag=sqrt(pow/(2*oran))*randn(1,1024);

      randn('state',4*(i+j));
      noi2_real=sqrt(pow/(2*oran))*randn(1,1024);

      randn('state',5*(i+j));
      noi2_imag=sqrt(pow/(2*oran))*randn(1,1024);

      noi1=noi1_real+j*noi1_imag;
      noi2=noi2_real+j*noi2_imag;
```

96

```
        xn=x+noi1; % x + noise
        yn=y+noi2; % y + noise

        % TDOA CALCULATION WITHOUT DENOISING

        xy=xcorr(xn,yn);
        [a1 b1]=max(real(xy));

        er1(i)=delay(i)-(b1-1024);

        %Time varying MAML

        e1=aml2(real(xn),real(yn),delay(i));
        e2=aml2(imag(xn),imag(yn),delay(i));

        er8(i)=(e1+e2)/2;

    end

    error8t(k)=(1/length(er8))*sum(er8.^2);
    error1a(k)=(1/length(er1))*sum(er1.^2);

    H8t(k,:)=er8;
    H1a(k,:)=er1;

end
load error8a

figure(6)
k=[5 4 3 2 1 0 -1 -2 -3 -4];
plot(k,error1a(1:10),'o',k,error8t(1:10),'x',k,error8a(1:10),'d',...
     k,error1a(1:10),k,error8t(1:10),k,error8a(1:10))

legend('xcorr without denoising','time varying aml','aml')

save error8t;
save error1a;

save H1a;
save H8t;
```

3.   Wavelet Denoising Based On The Fourth Order Moment

```
%*************************************************************
%*************************************************************
% sta: Wavelet Denosing Based on The Fourth Order Moment
%
% SYNTAX: y=sta(xn,yn,delay)
%
% INPUT: xn = Received signal from first receiver
%        yn = Received signal from second receiver
%     delay = True TDOA between two xn and yn
%
% OUTPUT: y = Error between true TDOA and estimated TDOA
%
% SUB_FUNC: None                                        .
% Written by Unal Aktas
%*************************************************************
%*************************************************************


function y=sta(xn,yn,delay);

xyn=xcorr(xn,yn,'biased');
[sigmas b]=max(xyn);
rx=xcorr(xn,'biased');
maxx=rx(length(xn));
ry=xcorr(yn,'biased');
maxy=ry(length(yn));
sigman1=maxx-sigmas;
sigman2=maxy-sigmas;

lamdax=3.1*sigman1^2;
lamday=3.1*sigman2^2;

nx=floor(log2(length(xn)));
ny=floor(log2(length(yn)));

[cx lx]=wavedec(xn,nx,'db4');
[cy ly]=wavedec(yn,ny,'db4');

dxc=[];
for i=1:nx
   d=detcoef(cx,lx,i);
   dl=length(d);
   A=(1/dl)*sum(d.^4);

   if A<lamdax
      dc=zeros(1,dl);
   else
      dc=d;
   end

   dxc=[dc dxc];
end

a=appcoef(cx,lx,'db4',nx);
al=length(a);
```

```
A=(1/al)*sum(a.^4);

if A<lamdax
    ac=zeros(1,al);
else
    ac=a;
end

dxc=[ac dxc];

xd=waverec(dxc,lx,'db4');

dyc=[];
for i=1:ny
    dy=detcoef(cy,ly,i);
    dyl=length(dy);
    A=(1/dyl)*sum(dy.^4);

    if A<lamday
        dcy=zeros(1,dyl);
    else
        dcy=dy;
    end

    dyc=[dcy dyc];
end

ay=appcoef(cy,ly,'db4',ny);
ayl=length(ay);
A=(1/ayl)*sum(ay.^4);

if A<lamday
    acy=zeros(1,ayl);
else
    acy=ay;
end

dyc=[acy dyc];

yd=waverec(dyc,ly,'db4');

rxyd=xcorr(xd,yd);

[a5 b5]=max(rxyd);

er5=delay-(b5-1024);
y=er5;
```

```
%***************************************************************************
%***************************************************************************
% tez6: This is a test program for wavelet denosing
%        based on the fourth order moment technique.
%        GSM signal is used.
%
% SYNTAX: tez6
%
% INPUT: None
%
% OUTPUT: Mean square error versus SNR
%
% SUB_FUNC: sta
% Written by Unal Aktas
%***************************************************************************
%***************************************************************************
clear all

gsm_set

data=data_gen(INIT_L);
[tx_burst,I,Q]=gsm_mod(Tb,OSR,BT,data,TRAINING);

s=I+j*Q;
sl=length(s);
pow=(1/sl)*sum(abs(s).^2);

K=200
f=150*rand(1,K);
delay=floor(f);
delay(1:K/2)=-delay(1:K/2);
n=[5 4 3 2 1 0 -1 -2 -3 -4 -5 -6];
SNR=10.^(n./10);

for k=1:length(SNR)

    oran=SNR(k)
    for i=1:K

        x=[zeros(1,200) s zeros(1,224)];
        y=[zeros(1,200+delay(i)) s zeros(1,224-delay(i))];

        randn('state',2*(i+j));
        noi1_real=sqrt(pow/(2*oran))*randn(1,1024);

        randn('state',3*(i+j));
        noi1_imag=sqrt(pow/(2*oran))*randn(1,1024);

        randn('state',4*(i+j));
        noi2_real=sqrt(pow/(2*oran))*randn(1,1024);

        randn('state',5*(i+j));
        noi2_imag=sqrt(pow/(2*oran))*randn(1,1024);

        noi1=noi1_real+j*noi1_imag;
        noi2=noi2_real+j*noi2_imag;
```

```
        xn=x+noi1;
        yn=y+noi2;

        % TDOA CALCULATION WITHOUT DENOISING

        xy=xcorr(xn,yn);
        [a1 b1]=max(real(xy));

        er1(i)=delay(i)-(b1-1024);

        %WAVELET DENOISING BASED ON THE FOURTH ORDER MOMENT

        e1=sta(real(xn),real(yn),delay(i));
        e2=sta(imag(xn),imag(yn),delay(i));

        er10(i)=(e1+e2)/2;

    end

    error10a(k)=(1/length(er10))*sum(er10.^2);
    error1a(k)=(1/length(er1))*sum(er1.^2);

    H10a(k,:)=er10;
    H1a(k,:)=er1;
end
figure(6)
k=[5 4 3 2 1 0 -1 -2 -3 -4];
plot(k,error1a(1:10),'o',k,error10a(1:10),'x',k,error1a(1:10),k,error10
a(1:10))
legend('xcorr without denoising','Fourth Order Moment')

save error10a;
save error1a;

save H1a;
save H10a;
```

LIST OF REFERENCES

1. *FCC Report and Order and Further Notice of Proposed Rule Making,* FCC Docket No. 94-102, July 1996

2. Caffery, J. J. and Stuber, G. L., "Radio Location in Urban CDMA Microcells," *Sixth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications,* pp 858-862, September 1995

3. Loomis, H. H., "Geolocation of Electromagnetic Emitters," October 3, 1996

4. Norre, E. A. and Jan, M. H., "GSMsim, A Matlab Implementation of a GSM Simulation Platform," December 1997

5. Rappaport, T. S., *Wireless Communications,* Prentice Hall, Upper Saddle River, NJ, 1996

6. Polikar, R., *The Wavelet Tutorial*

7. *The Matlab Wavelet Toolbox,* The Mathworks, Inc., Natick, MA, 1996

8. Haykin, S., *An Introduction to Analog and Digital Communications,* John Wiley & Sons, Inc., New York, 1989

9. Barsanti, R., *Denoising of Ocean Acoustic Signals Using Wavelet-Based Techniques,* Master's Thesis, Naval Postgraduate School, Monterey, CA, December 1996

10. Burrus, C.S., Gopinath, R.A., and Guo, H., *Introduction To Wavelets And Wavelet Transforms,* Prentice Hall, Upper Saddle River, NJ, 1998

11. Chambers, E.G., *Statistical Calculation for Beginners,* Cambridge, MA, 1958

12. Proakis, J.G., and Manolokis, D.G., *Digital Signal Processing Principles, Algorithms and Aplications,* Prentice Hall, Upper Saddle River, NJ, 1996

13. Rappaport, T. S., Reed, J. H., and Woemer, B. D., "Position Location Using Wireless Communication on Highways of the Future," *IEEE Communications Magazine,* pp 33-41, October 1996

14. *The Matlab Signal Processing Toolbox,* The Mathworks, Inc., Natick, MA, February, 1994

15. Wo, S. Q., So, H. C., and Ching, P. C., "Improvement of TDOA Measurement Using Wavelet Denoising with a Novel Thresholding Technique," *IEEE Acoustic, Speech and Signal Processing*, pp 539-542, April 1997

16. Wong, R. S. C., "Denoising of Low-SNR Signals Using Composite Wavelet Shrinkage," *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, pp 302-305, August 1997

17. Chan, Y. T., So, H. C. and Ching, P. C., "Approximate Maximum-Likelihood Delay Estimation via Orthogonal Wavelet Transform," *IEEE International Symposium on Circuits and Systems*, pp 2501-2504, June 9-12, 1997, Hong Kong

18. Donoho, D. and Johnstone I., "Ideal Spatial Adaptation via Wavelet Shrinkage," *Biometrika*, vol. 81, pp 425-455, 1994

19. Donoho, D., "Denoising by Thresholding," *IEEE Information Theory*, vol. 41, pp 613-627, May 1995

20. Donoho, D. and Johnstone, I., "Adapting to Unknown Smoothness via Wavelet Shrinkage," *Journal of American Statistics Assoc.*, vol. 90, pp 1200-1224, December 1995

21. Stein, C., "Estimation of The Mean of a Multivariate Normal Distribution," *The Annals of Statistics*, vol. 9, pp 1135-1151, 1981

22. Harris, F. J., "On the Use of Windows for Harmonic Analysis With the Discrete Fourier Transform," *IEEE Proceedings*, Vol. 66, pp 51-83, January 1978

23. Getting, I. A., "Global Positioning Systems," *IEEE Spectrum*, pp 36-47, December 1993

24. Rappaport, T. S., *Wireless Communications: Principle & Practice*, Upper Saddle River, NJ, Prentice Hall, 1996

25. Zhen, R., "Capacity and Interference Resistance of Spread Spectrum Automatic Vehicle Monitoring Systems in 902-928 MHz Band," Virginia Tech, Mobile and Portable Radio Research Group-TR-94-26, September 1994

INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Technical Information Center....................................2
   8725 John J. Kingman Rd., STE 0944
   Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library...2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, CA 93943-5101

3. Chairman, Code EC...1
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5121

4. Prof. Ralph D. Hippenstiel, Code EC/Hi.....................................2
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5121

5. Prof. Tri T. Ha, Code EC/Ha..1
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5121

6. Prof. H. H. Loomis, Code EC/Lm...1
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5121

7. Deniz Kuvvetleri Komutanligi...2
   Personel Daire Baskanligi
   Bakanliklar
   Ankara, TURKEY

8. Deniz Harp Okulu Komutanligi...1
   Kutuphane
   Tuzla, Istanbul- 81704, TURKEY

    Adil Demir Sitesi
    16. Blok, Daire no 2
    Bornova/Izmir, TURKEY